SIAM J. IMAGING SCIENCES
Vol. 8, No. 3, pp. 1489–1518

Video Compressive Sensing for Spatial Multiplexing Cameras Using Motion-Flow Models*

Aswin C. Sankaranarayanan†, Lina Xu‡, Christoph Studer§, Yun Li‡, Kevin F. Kelly‡, and Richard G. Baraniuk‡

Abstract. Spatial multiplexing cameras (SMCs) acquire a (typically static) scene through a series of coded projections using a spatial light modulator (e.g., a digital micromirror device) and a few optical sensors. This approach finds use in imaging applications where full-frame sensors are either too expensive (e.g., for short-wave infrared wavelengths) or unavailable. Existing SMC systems reconstruct static scenes using techniques from compressive sensing (CS). For videos, however, existing acquisition and recovery methods deliver poor quality. In this paper, we propose the CS multiscale video (CS-MUVI) sensing and recovery framework for high-quality video acquisition and recovery using SMCs. Our framework features novel sensing matrices that enable the efficient computation of a low-resolution video preview, while enabling high-resolution video recovery using convex optimization. To further improve the quality of the reconstructed videos, we extract optical-flow estimates from the low-resolution previews and impose them as constraints in the recovery procedure. We demonstrate the efficacy of our CS-MUVI framework for a host of synthetic and real measured SMC video data, and we show that high-quality videos can be recovered at roughly 60× compression.

Key words. video compressive sensing, optical flow, measurement matrix design, spatial multiplexing cameras

AMS subject classifications. 68U10, 68T45

DOI. 10.1137/140983124

1. Introduction.

Compressive sensing (CS) enables one to sample signals that admit a sparse representation in some transform basis well-below the Nyquist rate, while still enabling their faithful recovery [3, 7]. Since many natural and man-made signals exhibit sparse representations, CS has the potential to reduce the costs associated with sampling in numerous practical applications.

1.1. Spatial multiplexing cameras.

The single pixel camera (SPC) [8] and its multipixel extensions [6, 21, 38] are spatial multiplexing camera (SMC) architectures that rely on CS. In this paper, we focus on such SMC designs, which acquire random (or coded) projections of a (typically static) scene using a spatial light modulator (SLM) in combination with a

*Received by the editors November 4, 2014; accepted for publication (in revised form) May 27, 2015; published electronically July 23, 2015.
http://www.siam.org/journals/siims/8-3/98312.html

†Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 (saswin@ece.cmu.edu). This author was supported by NSF grant CCF-1117939.

‡Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 (lx2@rice.edu, yun.li@rice.edu, kkelly@rice.edu, richb@rice.edu). The second, fourth, and fifth authors were supported by ONR (N660011114090), DARPA KeCoM (11DARPA1055) through Lockheed Martin, and Princeton MIRTHE (NSF EEC 0540832). The sixth author was supported by NSF grants CCF-0431150, CCF-0728867, CCF-0926127, CCF-1117939, ARO MURI W911NF-09-1-0383, W911NF-07-1-0185, DARPA N66001-11-1-4090, N66001-11-C-4092, N66001-08-1-2065, ONR N00014-12-1-0124, and AFOSR FA9550-09-1-0432.

§Department of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853 (studer@cornell.edu).

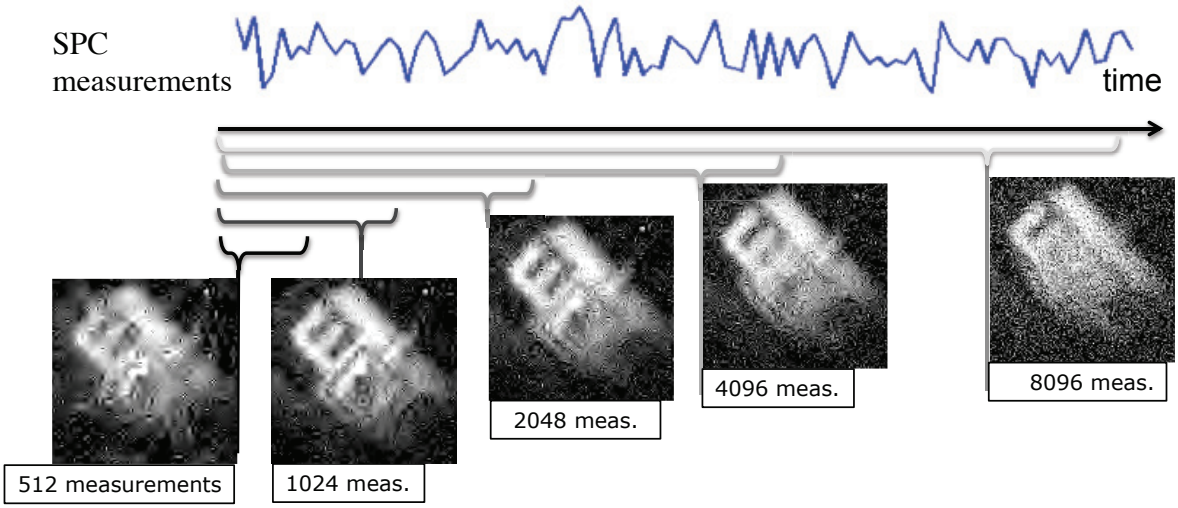


Figure 1. SPC and the static scene assumption. *An SPC acquires a single measurement per time-instant. If the scene were static, one could aggregate multiple measurements over time to recover the image of the scene via sparse signal recovery; for dynamic scenes, however, this approach fails. Shown above are reconstructs of a scene comprising a pendulum with the letter "R," swinging from right to left. We show reconstructed images using different numbers of aggregated (or grouped) measurements. Aggregating only a small number of measurements results in poor image quality. Aggregating a large number of measurements violates the static scene assumption and results in dramatic temporal aliasing artifacts.*

small number of optical sensors, such as single photodetectors or bolometers. The use of a small number of optical sensors—in contrast to full-frame sensors having millions of pixel elements—turns out to be advantageous when acquiring scenes at nonvisible wavelengths. Since the acquisition of scene information beyond the visual spectrum often requires sensors built from exotic materials, corresponding full-frame sensor devices are either too expensive or cumbersome [10].

Obviously, the use of a small number of sensors is, in general, not sufficient for acquiring complex scenes at high resolution. Hence, existing SMCs assume that the scenes to be acquired are static and acquire multiple measurements over time. For static scenes (i.e., images) and for a single pixel SMC architecture, this sensing strategy has been shown to deliver good results [8] typically at a compression of 2–8×. This approach, however, fails for time-variant scenes (i.e., videos). The main reason is due to the fact that the time-varying scene to be captured is ephemeral, i.e., *each* measurement acquires information of a (slightly) *different* scene. The situation is further aggravated when we deal with SMCs having a very small number of sensors (e.g., only one for the SPC). Virtually all existing methods for CS-based video recovery (e.g., [22, 25, 32, 34, 37]) seem to overlook the important fact that scenes are changing while one acquires compressive measurements. In fact, all of the mentioned SMC video systems treat scenes as a sequence of *static* frames (i.e., as piecewise constant scenes) as opposed to a continuously changing scene. This disconnect between the real-world operation of SMCs and the assumptions commonly made for video CS motivates novel SMC acquisition systems and recovery algorithms that are able to deal with the ephemeral nature of real scenes. Figure 1 illustrates the effect of assuming piecewise static scenes. Put simply, grouping too

few measurements for reconstruction results in poor spatial resolution; grouping too many measurements results in severe temporal aliasing artifacts.

1.2. The "chicken-and-egg" problem of video CS. High-quality video CS recovery methods for camera designs relying on temporal multiplexing (in contrast to spatial multiplexing, as is the case for SMCs) are generally inspired by video compression schemes and exploit motion estimation between individually recovered frames [28]. Applying such techniques for SMC architectures, however, results in a fundamental problem. On the one hand, obtaining motion estimates (e.g., the optical flow between pairs of frames) requires knowledge of the individual video frames. On the other hand, recovering the video frames in the absence of motion estimates is difficult, especially when using low sampling rates and a small number of sensor elements (cf. Figure 1). Attempts to address this "chicken-and-egg" problem either perform multiscale sensing [25] or sense separate patches of the individual video frames [22]. However, both approaches ignore the time-varying nature of real-world scenes and rely on a piecewise static scene model.

1.3. The CS-MUVI framework. In this paper, we propose a novel sensing and recovery method for videos acquired by SMC architectures, such as the SPC [8]. We start (in section 3) with an overview of our sensing and recovery framework. In section 4, we study the recovery performance of time-varying scenes and demonstrate that the performance degradation caused by violating the static scene assumption is severe, even at moderate levels of motion. We then detail a novel video CS strategy for SMC architectures that overcomes the static scene assumption. Our approach builds upon a codesign of scene acquisition and video recovery. In particular, we propose a novel class of CS matrices that enables us to obtain a low-resolution "preview" of the scene at low computational complexity. This preview video is used to extract robust motion estimates (i.e., the optical flow) of the scene at full-resolution (in section 5). We exploit these motion estimates to recover the full-resolution video by using off-the-shelf convex-optimization algorithms typically used for CS (in section 6). We demonstrate the performance and capabilities of our SMC video recovery algorithm for different scenes in section 7, show video recovery on real data in section 8, and discuss our findings in section 9. Given the multiscale nature of our framework, we refer to it as CS multiscale video (CS-MUVI).

We note that a short version of this paper was presented at the IEEE International Conference on Computational Photography [31] and the Computational Optical Sensing and Imaging meeting [40]. This paper contains an improved recovery algorithm, a more detailed performance analysis, and a larger number of experimental results. Most important, we show—to the best of our knowledge—the first high-quality video recovery results from real data obtained with a laboratory SPC; see Figure 2 for corresponding results.

2. Background.

2.1. Design of multiplexing systems. Suppose that we have a signal acquisition system characterized by $\mathbf{y} = \mathbf{A}\mathbf{x}^* + \mathbf{e}$, where $\mathbf{x}^* \in \mathbb{R}^N$ is the signal to be sensed and $\mathbf{y} \in \mathbb{R}^N$ is the measurement obtained using the matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$. The entries a_{ij} of the measurement matrix $\mathbf{A} \in \mathbb{R}^{N\times N}$ are usually restricted to $a_{ij} \in [-1, +1]$. Given an invertible matrix $\mathbf{A}$, the recovery error associated with the least-squares estimate $\widehat{\mathbf{x}} = \mathbf{A}^{-1}\mathbf{y} = \mathbf{x}^* + \mathbf{A}^{-1}\mathbf{e}$ satisfies the

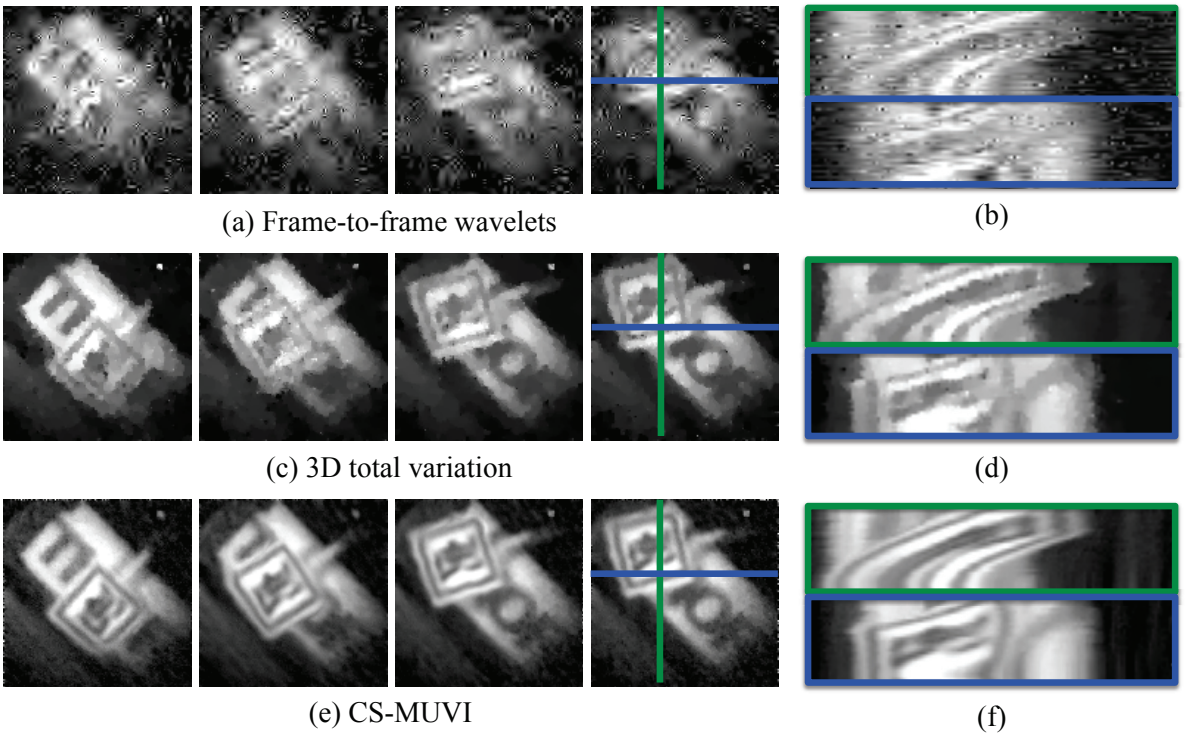


Figure 2. What a difference a signal model makes. *We show videos recovered from the same set of measurements but using different signal models:* (a) *sparsity of wavelet coefficients of individual frames of the video,* (b) *3D total variation enforcing sparse spatio-temporal gradients, and* (c) *CS-MUVI, the proposed video CS algorithm. The data were collected using an SPC operating in the short-wave infrared (SWIR) spectrum and acquiring* 10,000 *measurements/second at a spatial resolution of* 128×128 *pixels. The scene, similar to Figure* 1*, consists of a pendulum with the letter "R" swinging from right to left. A total of* 16,384 *measurements were acquired and videos were reconstructed under the three different signal models. Also shown are xt and yt slices corresponding to the lines marked. In all, CS-MUVI delivers high spatial as well as temporal resolution unachievable by both naive frame-to-frame wavelet sparsity as well as the more sophisticated 3D total variations model. To the best of our knowledge, CS-MUVI is the first demonstration of successful video recovery at* $128\times$ *super-resolution on* real data *obtained from an SPC.*

following inequality:

$$ERR(\widehat{\mathbf{x}}) = \|\widehat{\mathbf{x}} - \mathbf{x}^*\|_2 \leq \|\mathbf{A}^{-1}\|\|\mathbf{e}\|_2.$$

Traditional imaging systems mostly use the identity as the measurement matrix, i.e., $\mathbf{A} = \mathbf{I}_N$; such measurements result in an error equal to $\|\mathbf{e}\|_2$.

A classical problem is the design of matrix $\mathbf{A}$, which results in minimal recovery error. As shown in [14], Hadamard matrices are optimal in guaranteeing the smallest possible error when the measurement noise $\mathbf{e}$ is signal independent. Specifically, if an $N \times N$ Hadamard matrix were to exist, then the recovery error would satisfy $ERR(\widehat{\mathbf{x}}) \leq \|\mathbf{e}\|_2/\sqrt{N}$, which is a dramatic reduction from $ERR(\widehat{\mathbf{x}}) \leq \|\mathbf{e}\|_2$ achieved by $\mathbf{A} = \mathbf{I}_N$.

While Hadamard multiplexing provides immense benefits in the context of imaging, it still requires an invertible measurement matrix; i.e., the dimensionality of the measurement $\mathbf{y}$ needs to be the same as (or greater than) that of the sensed signal $\mathbf{x}^*$. For SMCs that

aggregate measurements over a time period, this implies a long acquisition period as the dimensionality of the signal N increases. This also leads to a poorer temporal resolution. All of these concerns could potentially be addressed if it were possible to reconstruct a signal from far fewer measurements than its dimensionality or when $M < N$. Such a sensing framework is popularly referred to as *compressive sensing*. We discuss this approach next.

2.2. Compressive sensing. CS deals with the estimation of a vector $\mathbf{x}^* \in \mathbb{R}^N$ from $M < N$ nonadaptive linear measurements [3, 7]

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}^* + \mathbf{e}, \tag{2.1}$$

where $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$ is the sensing matrix and $\mathbf{e}$ represents measurement noise. Estimating the signal $\mathbf{x}^*$ from the compressive measurements $\mathbf{y}$ is an ill-posed problem, in general, since the (noiseless) system of equations $\mathbf{y} = \mathbf{\Phi}\mathbf{x}^*$ is underdetermined. Early results in sparse polynomial interpolation [1] showed that, in the noiseless setting, it is possible to recover a K-sparse vector from $M = 2K$ measurements; however, the use of algebraic methods involving polynomials of high degree made the solutions fragile to perturbations. A fundamental result from CS theory states that a robust estimate of the vector $\mathbf{x}^*$ can be obtained from

$$M \sim K \log(N/K) \tag{2.2}$$

measurements if (i) the signal $\mathbf{x}^*$ admits a K-sparse representation $\mathbf{s}^* = \mathbf{\Psi}^T\mathbf{x}^*$ in an orthonormal basis $\mathbf{\Psi}$ (i.e., $\mathbf{s}^*$ has no more than K nonzero entries), and (ii) the *effective* sensing matrix $\mathbf{\Phi\Psi}$ satisfies the restricted isometry property (RIP) [2]. For example, if the entries of the sensing matrix $\mathbf{\Phi}$ are i.i.d. zero-mean Gaussian distributed, then $\mathbf{\Phi\Psi}$ is known to satisfy the RIP with high probability. Furthermore, any K-sparse signal $\mathbf{x}^*$ satisfying (2.2) can be estimated stably from the noisy measurement $\mathbf{y}$ by solving the following convex-optimization problem [3]:

$$\text{(P1)} \quad \widehat{\mathbf{x}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{\Psi}^T\mathbf{x}\|_1 \quad \text{subject to } \|\mathbf{y} - \mathbf{\Phi}\mathbf{x}\|_2 \le \epsilon.$$

Here, $(\cdot)^T$ denotes matrix transposition, and the parameter $\epsilon \ge \|\mathbf{e}\|_2$ is a bound on the measurement noise. For K-sparse signals, it can be shown that recovery error is bounded from above by $ERR(\widehat{\mathbf{x}}) \le C_0\epsilon$, where C_0 is a constant. Hence, in the noiseless setting (where $\epsilon = 0$), the K-sparse signal $\mathbf{x}^*$ can be recovered perfectly, even by acquiring far fewer measurements (2.2) than the signal's dimensionality.

Signals with sparse gradients. The results of CS have been extended to include a broad class of signals beyond that of sparse signals; an example of this are signals that exhibit sparse gradients. For such signals, one can solve problems of the form [5, 24]

$$\text{(TV)} \quad \widehat{\mathbf{x}} = \arg\min_{\mathbf{x}} \text{TV}(\mathbf{x}) \quad \text{subject to } \|\mathbf{y} - \mathbf{\Phi}\mathbf{x}\|_2 \le \epsilon,$$

where the gauge $\text{TV}(\mathbf{x})$ promotes sparse gradients. In the context of images where $\mathbf{x}$ denotes a 2D signal (i.e., an image), the operator $\text{TV}(\mathbf{x})$ can be defined as

$$\text{TV}_{\text{iso}}(\mathbf{x}) = \sum_i \sqrt{(D_x\mathbf{x}(i))^2 + (D_y\mathbf{x}(i))^2},$$

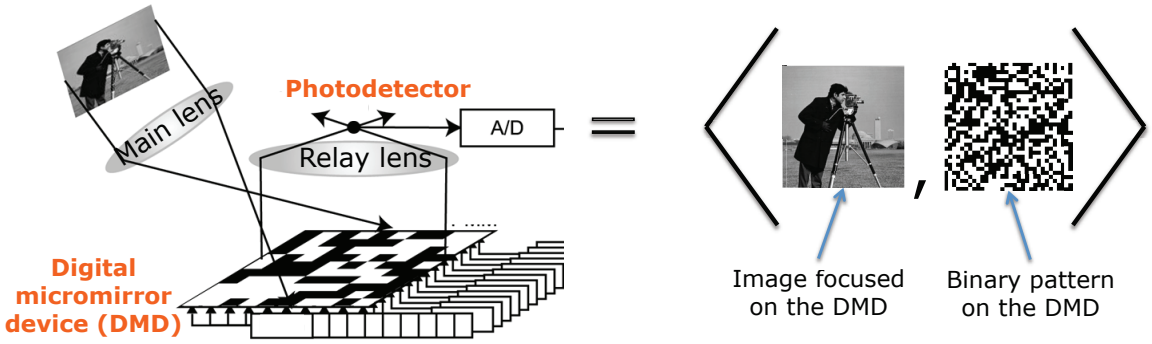


Figure 3. Operation principle of the SPC. *Each measurement is the inner-product between the binary mirror-orientation patterns on the DMD and the scene to be acquired.*

where $D_x\mathbf{x}$ and $D_y\mathbf{x}$ are the spatial gradients in the x- and the y-direction of the 2D image $\mathbf{x}$, respectively. This definition can easily be extended to higher-dimensional signals, such as RGB color images or videos (where the 3rd dimension is time). We next look at the prior art devoted specifically to CS of videos.

2.3. Video compressive sensing. An important challenge in CS of videos is that the temporal dimension is fundamentally different from spatial and spectral dimensions due to its ephemeral nature. The causality of time prevents us from obtaining additional measurements of an event that has already occurred. This is especially relevant for SMCs that aggregate measurements over a time period. Further, temporal statistics of a video are often different from the spatial statistics. These unique characteristics have led to a large body of work dedicated to video CS, which can be broadly grouped into signal models and corresponding recovery algorithms, and novel compressive imaging architectures.

2.3.1. Spatial multiplexing cameras. SMCs are imaging architectures that build on the ideas of CS. In particular, they employ an SLM, e.g., a digital micromirror device (DMD) or liquid crystal on silicon (LCOS), to optically compute a series of linear projections of the scene $\mathbf{x}$; these linear projections determine the rows of the sensing matrix $\mathbf{\Phi}$. Since SMCs are usually built with only a few sensor elements, they can operate at wavelengths where corresponding full-frame sensors are too expensive. In the recovery stage, one estimates the image $\mathbf{x}$ from the compressive measurements collected in $\mathbf{y}$, for example, by solving (P1) or variants thereof.

Single pixel camera. A prominent SMC is the SPC [8]; its main feature is the ability to acquire images using only a *single* sensor element (i.e., a single pixel) and by taking significantly fewer multiplexed measurements than the number of pixels of the scene to be recovered. In the SPC, light from the scene is focused onto a programmable DMD, which directs light from only a subset of activated micromirrors onto the photodetector. The programmable nature of the DMD enables us to freely direct light from each of the micromirrors towards the photodetector or away from it. As a consequence, the voltage measured at the photodetector corresponds to an inner-product of the image focused on the DMD and the activation pattern of the DMD (see Figure 3). Specifically, at time t, if the DMD pattern were ϕ_t and the scene were $\mathbf{x}_t$,

then the photodetector would measure a scalar value $y_t = \langle \phi_t, \mathbf{x}_t \rangle + e_t$, where $\langle \cdot, \cdot \rangle$ denotes the inner-product between the vectors. If the scene were static $\mathbf{x}_t = \mathbf{x}$, then multiple measurements could be aggregated to form the expression in (2.1), with $\mathbf{\Phi} = [\phi_1,\ \phi_2, \ldots,\ \phi_M]^T$. The SPC leverages the high operating speed of the DMD; i.e., the mirror's orientation patterns on the DMD can be reprogrammed at kHz rates. The DMD's operating speed defines the measurement bandwidth (i.e., the number of measurements/second), which is one of the key factors that define the achievable spatial and temporal resolutions.

There have been many recovery algorithms proposed for video CS using the SPC. Wakin et al. [37] use 3D wavelets as a sparsifying basis for videos and recover all frames of the video jointly under this prior. Unlike images, videos are not well represented using wavelets since they have additional temporal properties, like brightness constancy, that are better represented using motion-flow models. Park and Wakin [26] analyzed the coupling between spatial and temporal bandwidths of a video. In particular, they argue that reducing the spatial resolution of a scene implicitly reduces its temporal bandwidth, and hence lowers the error caused by the static scene assumption. This builds the foundation for the multiscale sensing and recovery approach proposed in [25], where several compressive measurements are acquired at multiple scales for each video frame. The recovered video at coarse scales (low spatial resolution) is used to estimate motion, which is then used to boost the recovery at finer scales (high spatial resolution). Other scene models and recovery algorithms for video CS with the SPC use block-based models [9, 22], sparse frame-to-frame residuals [4, 35], linear dynamical systems [32, 33, 34], and low rank plus sparse models [39]. To the best of our knowledge, all of these report results only on synthetic data and work under the assumption that each frame of the video remains static for a certain duration of time (typically 1/30 of a second)—an assumption that is violated when operating with an actual SPC.

2.3.2. Temporal multiplexing cameras. In contrast to SMCs that use sensors having low spatial resolution and seek to spatially super-resolve images and videos, temporal multiplexing cameras (TMCs) have low frame rate sensors and seek to temporally super-resolve videos. In particular, TMCs use SLMs for temporal multiplexing of videos and sensors with high spatial resolution such that the intensity observed at each pixel is coded temporally by the SLM during each exposure.

Veeraraghavan, Reddy, and Raskar [36] showed that periodic scenes could be imaged at very high temporal resolutions by using a global shutter or a "flutter shutter" [27]. This idea was extended to nonperiodic scenes in [16] where a union-of-subspace model was used to temporally super-resolve the captured scene. Reddy, Veeraraghavan, and Chellappa [28] proposed the programmable pixel compressive camera (P2C2), which extends the flutter shutter idea with per-pixel shuttering. Inspired from video compression standards such as MPEG-1 [18] and H.264 [29], the recovery of videos from the P2C2 was achieved using the optical flow between pairs of consecutive frames of the scene. The optical flow between pairs of video frames is estimated using an initial reconstruction of the high frame rate video using wavelet priors on the individual frames. A second reconstruction is then performed that further enforces the brightness constancy expressions provided by the optical-flow fields. The implementation of the recovery procedure described in [28] is tightly coupled to the imaging architecture and prevents its use for SMC architectures. Nevertheless, the use of optical-flow estimates for

video CS recovery inspired the recovery stage of CS-MUVI as detailed in section 6.

Gu et al. [12] propose using the rolling shutter of a CMOS sensor to enable higher temporal resolution. The key idea there is to stagger the exposures of each row randomly and use image/video statistics to recover a high frame rate video. Hitomi et al. [15] use a per-pixel coding, similar to P2C2, that is implementable in modern CMOS sensors with per-pixel electronic shutters; however, a hallmark of their approach is the use of a highly overcomplete dictionary of video patches to recover the video at high frame rates. This results in highly accurate reconstructions even when brightness constancy—the key construct underlying optical flow estimation—is violated. Llull et al. [20] propose a TMC that uses a translating mask in the sensor plane to achieve temporal multiplexing. This approach avoids the hardware complexity involved with DMDs and LCOS and enjoys other benefits including low operational power consumption. In Yang et al. [42], a Gaussian mixture model (GMM) is used as a signal prior to recovery high frame rate videos for TMCs; a hallmark of this approach is that the GMM parameters are not just trained offline but also adapted and tuned in situ during the recovery process. Harmany, Marcia, and Willett [13] extend coded aperture systems by incorporating a flutter shutter [27] or a coded exposure; the resulting TMC provides immense flexibility in the choice of measurement matrix. They also show the resulting system provides measurement matrices that satisfy the RIP.

3. Overview of CS-MUVI.

State-of-the-art video compression methods rely on estimating the motion in the scene, compress a few reference frames, and use the motion vectors that relate the remaining parts of a scene to these reference frames. While this approach is possible in the context of video compression, i.e., where the algorithm has prior access to the entire video, it is significantly more difficult in the context of compressive sensing.

A general strategy to enable the use of motion flow–based signal models for video CS is to use a two-step approach [28]. In the first step, an initial estimate of the video is generated by recovering each frame individually using sparse wavelet or gradient priors. The initial estimate is used to derive motion flow between consecutive frames; this enables a powerful description in terms of relating intensities at pixels across frames. In the second step, the video is re-estimated, but now with the aid of enforcing the extracted motion-flow constraints in addition to the measurement constraints. The success of this two-step strategy critically depends on the ability to obtain reliable motion estimates, which, in turn, depends on obtaining robust initial estimates in the first step. Unfortunately, in the context of SMCs, obtaining reliable initial estimates of the frames of the video, in the absence of motion knowledge, is inherently hard due to the violation of the static scene model (recall Figure 1).

The proposed framework, referred to as CS-MUVI, enables a robust initial estimate by obtaining the individual frames at a *lower spatial resolution*. This approach has two important benefits towards reducing the violation of the static scene model. First, obtaining the initial estimate at a lower spatial resolution reduces the dimensionality of the video significantly. As a consequence, we can estimate individual frames of the video from *fewer* measurements. In the context of an SMC, this implies a *smaller* time window over which these measurements are obtained, and hence, *reduced* misfit to the static scene model. Second, spatial downsampling naturally reduces the temporal resolution of the video [26]; this is a consequence of the additional blur due to spatial downsampling. This implies that the violation of the static

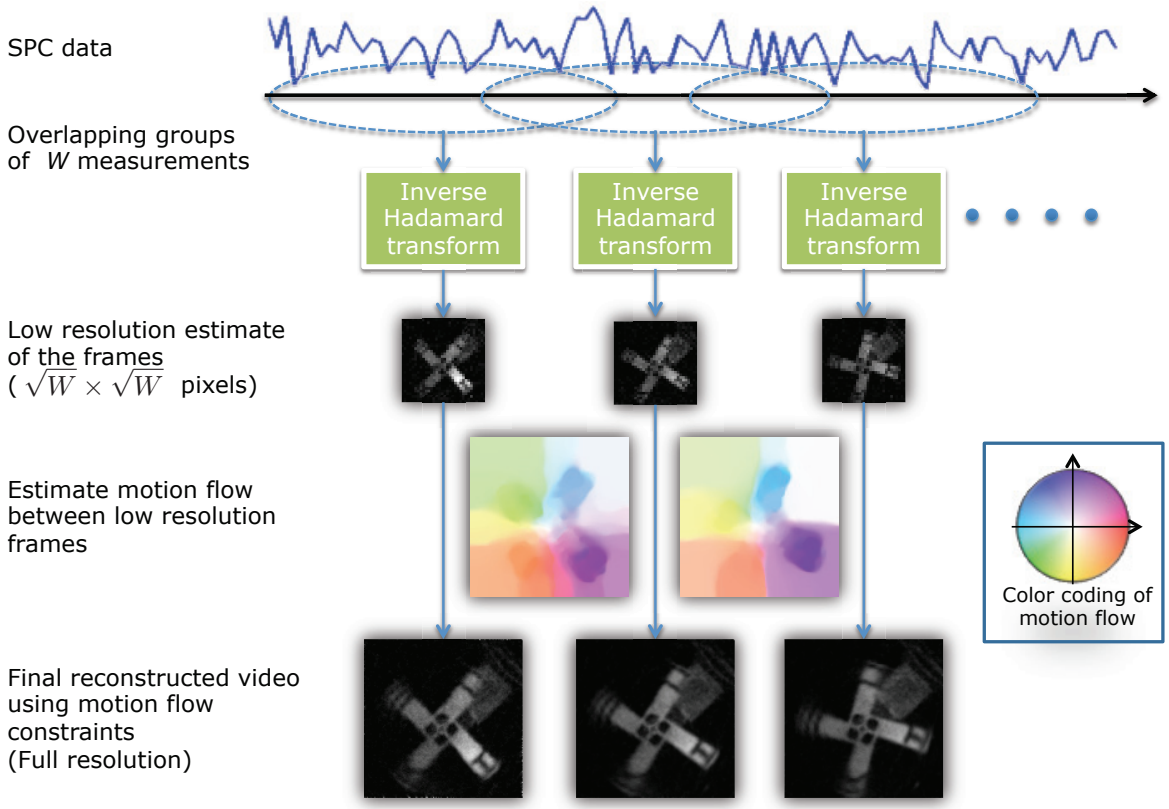


Figure 4. Outline of the CS-MUVI recovery framework. *Given a total number of T measurements, we group them into overlapping windows of size W resulting in a total of F frames. For each frame, we first compute a low-resolution initial estimate using a window of W neighboring measurements. We then compute the optical flow between upsampled preview frames (the optical flow is color-coded as in [19]). Finally, we recover F high-resolution video frames by enforcing a sparse gradient prior along with the measurement constraints, as well as the brightness constancy constraints generated from the optical-flow estimates.*

scene assumption is naturally *reduced* when the video is downsampled. In section 4, we study this strategy in detail and characterize the error in estimating the initial estimates at a lower resolution. Specifically, given W consecutive measurements from an SMC, we are interested in estimating a *single static* image at a resolution of $\sqrt{W} \times \sqrt{W}$ pixels. Note that varying W, which denotes the window length, varies both the spatial resolution of the recovered frame (since it has a resolution of $\sqrt{W} \times \sqrt{W}$) as well as its temporal resolution (since the acquisition time is proportional to W). We analyze various sources of error in the recovered low-resolution frame. This analysis provides conditions for stable recovery of the initial estimates that lead to the design of measurement matrices in section 5.

The proposed CS-MUVI framework for video CS relies on three steps. First, we recover a low-resolution video by reconstructing each frame of the video, individually, using simple least-squares techniques. Second, this low-resolution video is used to obtain motion estimates between frames. Third, we recover a high-resolution video by enforcing a spatio-temporal gradient prior, with the constraints induced by the compressive measurements as well as the constraints due to motion estimates. Figure 4 provides a schematic overview of these steps.

4. Spatio-temporal trade-off. We now study the recovery error that results from the static scene assumption while sensing a time-varying scene (video) with an SMC. We also identify a fundamental trade-off underlying a multiscale recovery procedure, which is used in section 5 to identify novel sensing matrices that minimize the spatio-temporal recovery errors. Since the SPC is the most challenging SMC architecture, as it only provides a single pixel sensor, we solely focus on the SPC in the following. Generalizing our results to other SMC architectures with more than one sensor is straightforward.

4.1. SMC acquisition model. The compressive measurements $y_t \in \mathbb{R}$ taken by a single pixel SMC at the sample instants $t = 1, \dots, T$ can be modeled as

$$y_t = \langle \phi_t, \mathbf{x}_t \rangle + e_t,$$

where T is the total number of acquired samples, $\phi_t \in \mathbb{R}^{N \times 1}$ is the measurement vector, $e_t \in \mathbb{R}$ represents measurement noise, and $\mathbf{x}_t \in \mathbb{R}^{N \times 1}$ is the scene (or frame) at sample instant t. In the remainder of the paper, we assume that the 2D scene consists of $n \times n$ spatial pixels, which, when vectorized, results in the vector $\mathbf{x}_t$ of dimension $N = n^2$. We also use the notation $\mathbf{y}_{1:W}$ to represent the vector consisting of a window of $W \leq T$ successive compressive measurements (samples), i.e.,

$$\mathbf{y}_{1:W} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_W \end{bmatrix} = \begin{bmatrix} \langle \phi_1, \mathbf{x}_1 \rangle + e_1 \\ \langle \phi_2, \mathbf{x}_2 \rangle + e_2 \\ \vdots \\ \langle \phi_W, \mathbf{x}_W \rangle + e_W \end{bmatrix}. \tag{4.1}$$

4.2. Static scene and downsampling errors. Suppose that we rewrite our (time-varying) scene $\mathbf{x}_t$ for a window of W consecutive sample instants as follows:

$$\mathbf{x}_t = \mathbf{b} + \Delta \mathbf{x}_t, \quad t = 1, \dots, W.$$

Here, $\mathbf{b}$ is the static component (assumed to be invariant for the considered window of W samples) and $\Delta \mathbf{x}_t = \mathbf{x}_t - \mathbf{b}$ is the error at sample instant t caused by the static scene assumption. By defining $z_t = \langle \phi_t, \Delta \mathbf{x}_t \rangle$, we can rewrite (4.1) as

$$\mathbf{y}_{1:W} = \mathbf{\Phi} \mathbf{b} + \mathbf{z}_{1:W} + \mathbf{e}_{1:W}, \tag{4.2}$$

where $\mathbf{\Phi} \in \mathbb{R}^{W \times N}$ is the sensing matrix whose tth row corresponds to the transposed measurement vector ϕ_t.

We now investigate the error caused by spatial downsampling of the static component $\mathbf{b}$ in (4.2). To this end, let $\mathbf{b}_L \in \mathbb{R}^{N_L}$ be the downsampled static component, and assume $N_L = n_L \times n_L$ with $N_L < N$. By defining a linear upsampling and downsampling operator as $\mathbf{U} \in \mathbb{R}^{N \times N_L}$ and $\mathbf{D} \in \mathbb{R}^{N_L \times N}$, respectively, we can rewrite (4.2) as follows:

$$\begin{aligned} \mathbf{y}_{1:W} &= \mathbf{\Phi}(\mathbf{U}\mathbf{b}_L + \mathbf{b} - \mathbf{U}\mathbf{b}_L) + \mathbf{z}_{1:W} + \mathbf{e}_{1:W} \\ &= \mathbf{\Phi}\mathbf{U}\mathbf{b}_L + \mathbf{\Phi}(\mathbf{b} - \mathbf{U}\mathbf{b}_L) + \mathbf{z}_{1:W} + \mathbf{e}_{1:W} \\ &= \mathbf{\Phi}\mathbf{U}\mathbf{b}_L + \mathbf{\Phi}(\mathbf{I} - \mathbf{U}\mathbf{D})\mathbf{b} + \mathbf{z}_{1:W} + \mathbf{e}_{1:W} \end{aligned} \tag{4.3}$$

since $\mathbf{b}_L = \mathbf{D}\mathbf{b}$. Inspection of (4.3) reveals three sources of error in the CS measurements of the low-resolution static scene $\mathbf{\Phi}\mathbf{U}\mathbf{b}_L$: (i) The *spatial-approximation error* $\mathbf{\Phi}(\mathbf{I} - \mathbf{U}\mathbf{D})\mathbf{b}$ caused by downsampling, (ii) the *temporal-approximation error* $\mathbf{z}_{1:W}$ caused by assuming the scene remains static for W samples, and (iii) the *measurement error* $\mathbf{e}_{1:W}$. Note that when $W \geq N_L$, the matrix ΦU has at least as many rows as columns, and hence we can get an estimate of $\mathbf{b}_L = (\Phi U)^\dagger \mathbf{y}_{1:W}$. We next study the error induced by this least-squares estimate in terms of the relative contributions of the spatial-approximation and temporal-approximation terms.

4.3. Estimating a low-resolution image. In order to analyze the trade-off that arises from the static scene assumption and the downsampling procedure, we consider the scenario where the effective matrix $\mathbf{\Phi}\mathbf{U}$ is of dimension $W \times N_L$ with $W \geq N_L$; that is, we aggregate at least as many compressive samples as the downsampled spatial resolution. If $\mathbf{\Phi}\mathbf{U}$ has full (column) rank, then we can obtain a least-squares estimate $\widehat{\mathbf{b}}_L$ of the low-resolution static scene $\mathbf{b}_L$ from (4.3) as

$$\widehat{\mathbf{b}}_L = (\mathbf{\Phi}\mathbf{U})^\dagger \mathbf{y}_{1:W} = \mathbf{b}_L + (\mathbf{\Phi}\mathbf{U})^\dagger \big(\mathbf{\Phi}(\mathbf{I} - \mathbf{U}\mathbf{D})\mathbf{b} + \mathbf{e}_{1:W} + \mathbf{z}_{1:W}\big), \tag{4.4}$$

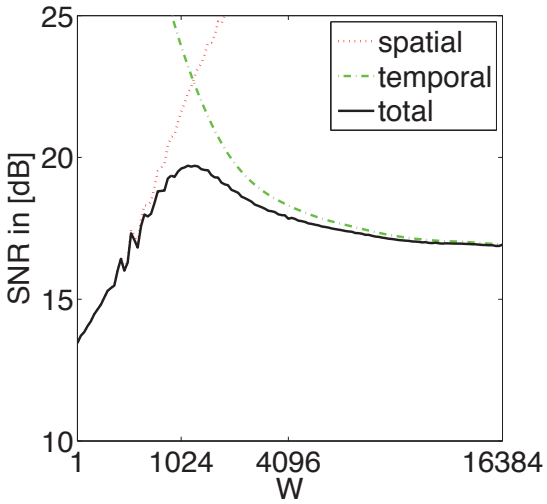
where $(\cdot)^\dagger$ denotes the pseudoinverse. From (4.4) we observe the following facts: (i) The window length W controls a trade-off between the spatial-approximation error $\mathbf{\Phi}(\mathbf{I} - \mathbf{U}\mathbf{D})\mathbf{b}$ and the error $\mathbf{z}_{1:W}$ induced by assuming a static scene $\mathbf{b}$ and (ii) the least-squares estimator matrix $(\mathbf{\Phi}\mathbf{U})^\dagger$ (potentially) amplifies all three error sources.

4.4. Characterizing the trade-off. The spatial-approximation error and the temporal-approximation error are both functions of the window length W. We now show that carefully selecting W minimizes the combined spatial and temporal error in the low-resolution estimate $\widehat{\mathbf{b}}_L$. A close inspection of (4.4) shows that for $W = 1$, the temporal-approximation error is zero, since the static component $\mathbf{b}$ is able to perfectly represent the scene at each sample instant t. As W increases, the temporal-approximation error increases for time-varying scenes; simultaneously, increasing W reduces the error caused by downsampling $\mathbf{\Phi}(\mathbf{I} - \mathbf{U}\mathbf{D})\mathbf{b}$ (see Figure 5(b)). For $W \geq N$ there is no spatial-approximation error (as long as $\mathbf{\Phi}\mathbf{U}$ is invertible). Note that characterizing both errors analytically is, in general, difficult as they heavily depend on the scene under consideration.

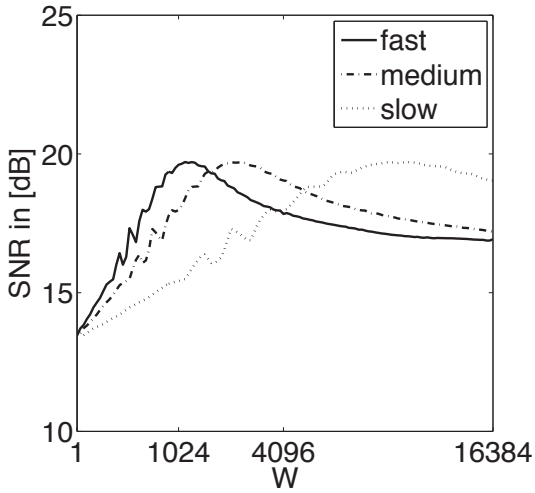
Figure 5 illustrates the trade-off controlled by W and the individual spatial- and temporal-approximation errors, characterized in terms of the recovery signal-to-noise ratio (SNR). The figure highlights our key observation that there is an optimal window length W for which the total recovery SNR is maximized. In particular, we see from Figure 5(c) that the optimum window length increases (i.e., towards higher spatial resolution) when the scene changes slowly; in contrast, when the scene changes rapidly, the window length (and consequently, the spatial resolution) should be low. Since $N_L \leq W$, the optimal window length W dictates the resolution for which accurate low-resolution motion estimates can be obtained. Hence, the optimal window length depends on the scene to be acquired, the rate at which measurements can be acquired, and the sensing matrix $\mathbf{\Phi}$ itself.

5. Design of sensing matrix. In order to bootstrap CS-MUVI, a low-resolution estimate of the scene is required. We next show that carefully designing the CS sensing matrix $\mathbf{\Phi}$

(a) Synthetic video of a translating object over a static textured background.



(b) Separate error sources.



(c) Impact of temporal changes.

Figure 5. Trade-off between spatial- and temporal-approximation errors. *The plots corresponding to a scene with a translating object over a static background.* (a) *Frames of a synthetic video with a spatial resolution of* 128×128 *pixels. The speed of movement of the cross is precisely controlled to subpixel accuracy.* (b) *The recovery SNRs caused by spatial- and temporal-approximation errors for values of* W*, the total number of measurements obtained. We collect* $W = n_L^2$ *measurements under the measurement model in* (4.1) *and reconstruct a single static frame* $\widehat{\mathbf{b}}_L$ *at a resolution of* $n_L \times n_L$ *such that* $(\mathbf{\Phi}U)$ *is invertible, using* (4.4)*. Next, since we have the ground truth, we can independently compute the spatial error* $\|\mathbf{b} - \mathbf{b}_L\|$ *as well as the temporal error* $\|\mathbf{z}_{1:W}\|$*.* (c) *We can vary the speed of motion of object and observe the dependence of the total approximation error on the speed of the object. At the medium speed, the cross translates so as to cover the field-of-view within* 16,384 *measurements; the speed of translation for the "slow" and "fast" motions corresponds to one-half and twice the speed of translation at "normal," respectively.*

enables us to compute high-quality low-resolution scene estimates at low complexity, which improves the performance of video recovery.

5.1. Dual-scale sensing matrices. The choice of the sensing matrix $\mathbf{\Phi}$ and the upsampling operator $\mathbf{U}$ are critical to arrive at a high-quality estimate of the low-resolution image $\mathbf{b}_L$. Indeed, if the effective matrix $\mathbf{\Phi U}$ is ill-conditioned, then application of the pseudoinverse

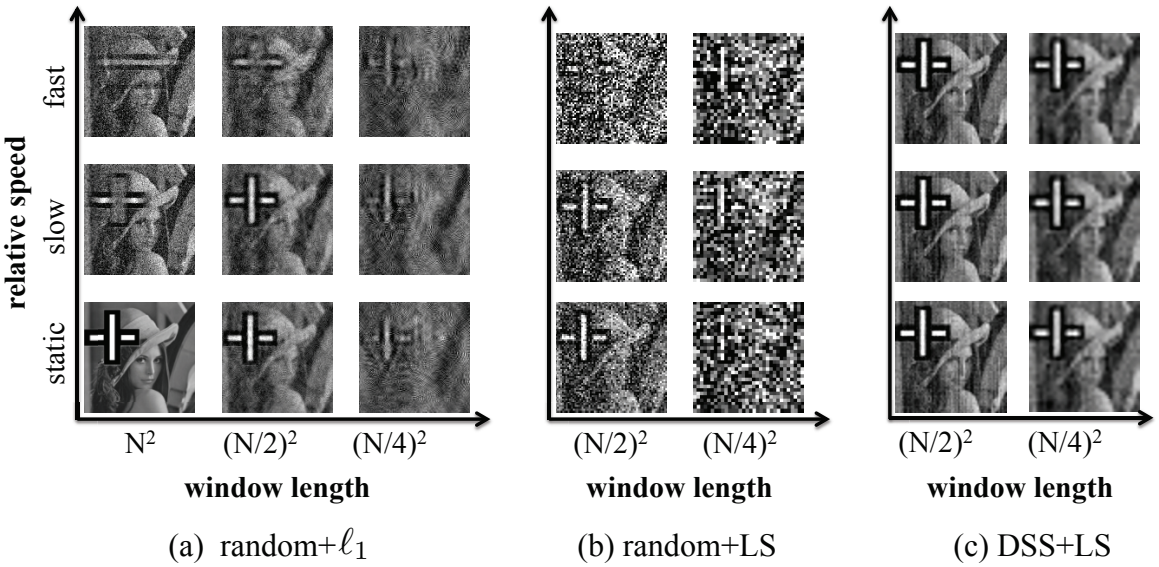


Figure 6. Performance of ℓ_1- and ℓ_2-based recovery algorithms for varying object motion. *The underlying scene corresponds to translating across a static background of Lena. The speed of translation of the cross is varied across different rows. Comparison between* (a) ℓ_1*-norm recovery,* (b) *least-squares recovery using a random matrix, and* (c) *least-squares recovery using a dual-scale sensing (DSS) matrix for various relative speeds (of the cross) and window lengths* W.

$(\mathbf{\Phi U})^\dagger$ amplifies all three sources of errors in (4.4), eventually resulting in a poor estimate. For virtually all sensing matrices $\mathbf{\Phi}$ commonly used in CS, such as i.i.d. (sub-)Gaussian matrices, as well as subsampled Fourier or Hadamard matrices, right multiplying them with an upsampling operator $\mathbf{U}$ often results in an ill-conditioned matrix or even a rank-deficient matrix. Hence, well-established CS matrices are a poor choice for obtaining a high-quality low-resolution preview. Figures 6(a) and 6(b) show recovery results for naïve recovery using (P1) and least-squares, respectively, using a random sensing matrix. We immediately see that both recovery methods result in poor performance, even for large window sizes W or for a small amount of motion.

In order to achieve good CS recovery performance *and* have minimum noise enhancement when computing a low-resolution preview $\widehat{\mathbf{b}}_L$ according to (4.4), we propose a novel class of sensing matrices, referred to as *dual-scale sensing* (DSS) matrices. These matrices will (i) satisfy the RIP to enable CS and (ii) remain well-conditioned when right multiplied by a given upsampling operator $\mathbf{U}$. Such a DSS matrix enables robust low-resolution as shown in Figure 6(c). We next discuss the details.

5.2. DSS matrix design.

In this section, we detail a particular design that is suited for SMC architectures. In SMC architectures, we are constrained in the choice of the entries of the sensing matrix $\mathbf{\Phi}$. Practically, the DMD limits us to matrices having binary-valued entries

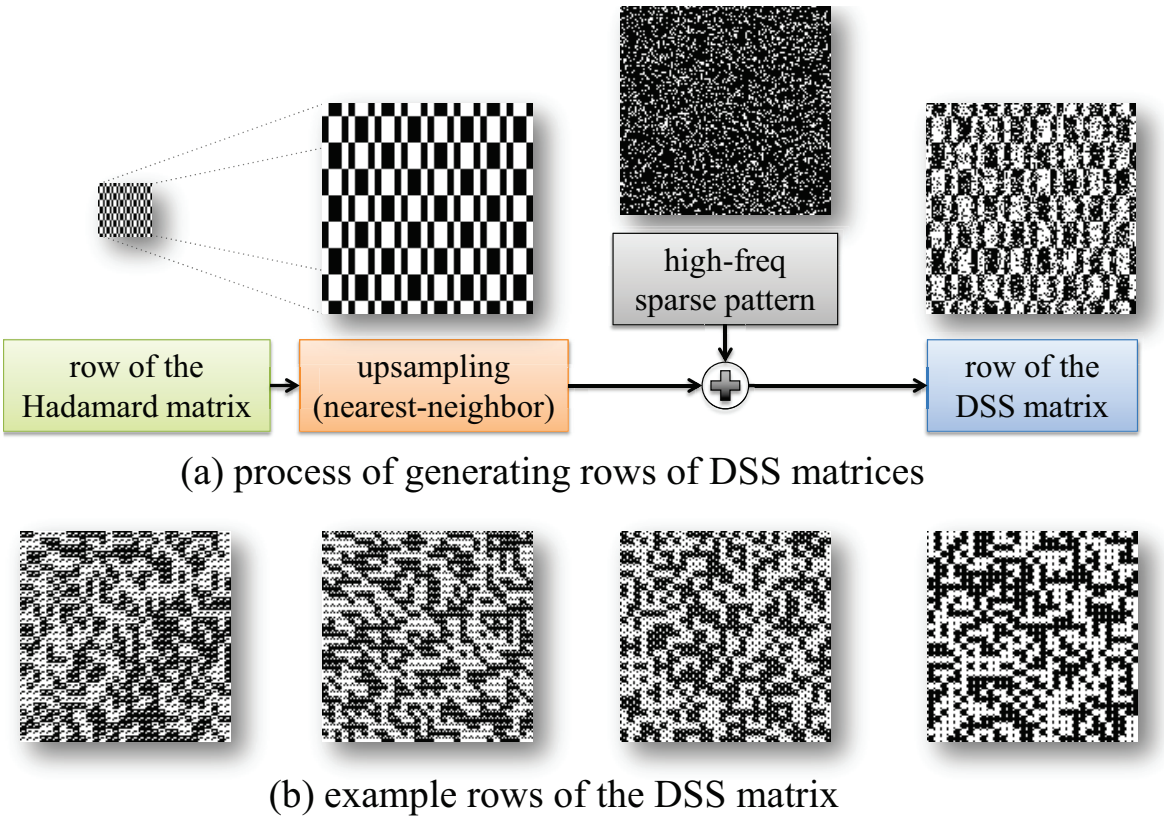


Figure 7. Generating DSS patterns. (a) *Outline of the process in* (5.1). (b) *In practice, we permute the low-resolution Hadamard for better incoherence with the sparsifying wavelet basis. Fast generation of the DSS matrix requires us to impose additional structure on the high-frequency patterns. In particular, each subblock of the high-frequency pattern is forced to be the same, which enables fast computation via convolutions.*

(e.g., ± 1) if we are interested in the highest possible measurement rate.[1] We propose the matrix $\mathbf{\Phi}$ to satisfy $\mathbf{H} = \mathbf{\Phi U}$, where $\mathbf{H}$ is a $W \times W$ Hadamard matrix[2] and $\mathbf{U}$ is a predefined upsampling operator. Recall from section 2.1 that Hadamard matrices have the following advantages: (i) they have orthogonal columns, (ii) they exhibit optimal SNR properties over matrices restricted to $\{-1, +1\}$ entries, and (iii) applying the (forward and inverse) Hadamard transform requires very low computational complexity (i.e., the same complexity as a fast Fourier transform).

We now show the construction of such a DSS matrix $\mathbf{\Phi}$ (see Figure 7(a)). A simple way to start is with a $W \times W$ Hadamard matrix $\mathbf{H}$ and to write the CS matrix as

$$\mathbf{\Phi} = \mathbf{HD} + \mathbf{F}, \tag{5.1}$$

where $\mathbf{D}$ is a downsampling matrix satisfying $\mathbf{DU} = \mathbf{I}$, and $\mathbf{F} \in \mathbb{R}^{W \times N}$ is an auxiliary matrix

[1] It is possible to employ more general sensing matrices, e.g., using spatial and/or temporal half-toning, which, however, comes at the cost of spatial resolution and/or speed. The design of such matrices is not within the scope of this paper, but an interesting research direction.

[2] In what follows, we assume that W is chosen such that a $W \times W$ Hadamard matrix exists.

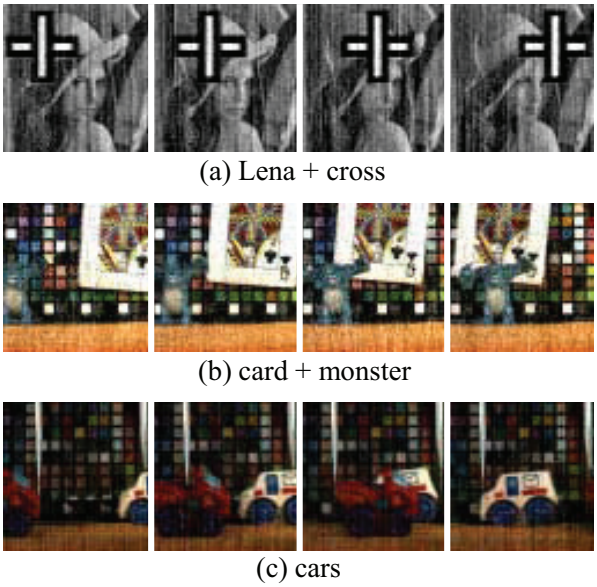


Figure 8. Preview frames for three different scenes. *All previews consist of* 64×64 *pixels. Preview frames are obtained at low computational cost using an inverse Hadamard transform, which opens up a variety of new real-time applications for video CS.*

that obeys the following constraints: (i) The entries of $\mathbf{\Phi}$ are ± 1, (ii) the matrix $\mathbf{\Phi}$ has good CS recovery properties (e.g., satisfies the RIP), and (iii) $\mathbf{F}$ should be chosen such that $\mathbf{FU} = \mathbf{0}$. Note that an easy way to ensure that $\mathbf{\Phi}$ is ± 1 is to interpret $\mathbf{F}$ as sign flips of the Hadamard matrix $\mathbf{H}$. Note that one could choose $\mathbf{F}$ to be an all-zeros matrix; this choice, however, results in a sensing matrix $\mathbf{\Phi}$ having poor CS recovery properties. In particular, such a matrix would inhibit the recovery of high spatial frequencies. Choosing random entries in $\mathbf{F}$ such that $\mathbf{FU} = \mathbf{0}$ (i.e., by using random patterns of high spatial frequency) provides excellent performance.

To arrive at an efficient implementation of CS-MUVI, we additionally want to avoid the storage of an entire $W \times N$ matrix. To this end, we generate each row $\mathbf{f}_i \in \mathbb{R}^N$ of $\mathbf{F}$ as follows. Associate each row vector $\mathbf{f}_i$ to an $n \times n$ image of the scene, partition the scene into blocks of size $(n/n_L) \times (n/n_L)$, and associate an $(n/n_L)^2$-dimensional vector $\hat{\mathbf{f}}_i$ with each block. We can now use the same vector $\hat{\mathbf{f}}_i$ for each block and choose $\hat{\mathbf{f}}_i$ such that the full matrix satisfies $\mathbf{FU} = \mathbf{0}$. We also permute the columns of the Hadamard matrix $\mathbf{H}$ to achieve better incoherence with the sparsifying bases used in section 6 (see Figure 7(b) for details).

5.3. Preview mode. The use of Hadamard matrices for the low-resolution part in the proposed DSS matrices has an additional benefit. Hadamard matrices have fast inverse transforms, which can significantly speed up the recovery of the low-resolution preview frames. Such a "fast" DSS matrix has the key capability of generating a high-quality *preview* of the scene (see Figure 8) with very low computational complexity; this is beneficial for video CS as it allows one to easily and quickly extract an estimate of the scene motion. The motion estimate can then be used to recover the video at its full resolution (see section 6). In addition

to this, the use of fast DSS matrices can be beneficial in various other ways, including (but not limited to) the following.

Digital viewfinder. Conventional SMC architectures do not enable the observation of the scene until CS recovery is performed. Due to the high computational complexity of most existing CS recovery algorithms, there is typically a large latency between the acquisition of a scene and its observation. Fast DSS matrices offer an *instantaneous* visualization of the scene, i.e., they can provide a real-time digital viewfinder; this capability substantially simplifies the setup of an SMC in practice.

Adaptive sensing. The immediate knowledge of the scene—even at a low resolution—is a key enabler for adaptive sensing strategies. For example, one may seek to extract the changes that occur in a scene from one frame to the next or track the locations of moving objects, while avoiding the typically high latency caused by computationally complex CS recovery algorithms.

5.4. Selecting W. Crucial to the design of the DSS matrix is the selection of the parameter W. While W is often scene-specific, a good rule of thumb is as follows: given an $n \times n$ scene, choose $W = n_L^2$ such that the motion of objects is less than n/n_L pixels in the amount of time required to get W measurements. Basically, this would serve to have motion in the preview images restricted to 1 pixel (at the resolution of the preview image).

6. Optical flow–based video recovery.

We next detail the second part of CS-MUVI, where we obtain the video at a high spatial resolution by estimating and enforcing motion estimates between frames.

6.1. Optical-flow estimation. Thanks to the preview mode, we can estimate the optical flow between any two (low-resolution) frames $\widehat{\mathbf{b}}_L^i$ and $\widehat{\mathbf{b}}_L^j$. For CS-MUVI, we compute optical-flow estimates at full spatial resolution between pairs of upsampled preview frames. For the results in this paper, we used "bicubic" interpolation to upsample the frames. This approach turns out to result in more accurate optical-flow estimates compared to an approach that first estimates the optical flow at low resolution followed by upsampling of the optical flow. Let $\widehat{\mathbf{b}}^i = \mathbf{U}\widehat{\mathbf{b}}_L^i$ be the upsampled preview frame. The optical-flow constraints between two frames, $\widehat{\mathbf{b}}^i$ and $\widehat{\mathbf{b}}^j$, can be written as

$$\widehat{\mathbf{b}}^i(x,y) = \widehat{\mathbf{b}}^j(x + u_{x,y}, y + v_{x,y}),$$

where $\widehat{\mathbf{b}}^i(x,y)$ denotes the pixel (x,y) in the $n \times n$ plane of $\widehat{\mathbf{b}}^i$, and $u_{x,y}$ and $v_{x,y}$ correspond to the translation of the pixel (x,y) between frames i and j (see [17, 19]).

In practice, the estimated optical flow may contain subpixel translations; i.e., $u_{x,y}$ and $v_{x,y}$ are not necessarily integer valued. If this is the case, then we approximate $\widehat{\mathbf{b}}^j(x+u_{x,y}, y+v_{x,y})$ as a linear combination of its four closest neighboring pixels,

$$\widehat{\mathbf{b}}^j(x + u_{x,y}, y + v_{x,y}) \approx \sum_{k,\ell \in \{0,1\}} w_{k,\ell} \widehat{\mathbf{b}}^j(\lfloor x + u_{x,y} \rfloor + k, \lfloor y + v_{x,y} \rfloor + \ell),$$

where $\lfloor \cdot \rfloor$ denotes rounding towards $-\infty$ and the weights $w_{k,\ell}$ are chosen according to the location within the four neighboring pixels. In order to obtain robustness against occlusions,

we enforce consistency between the forward and backward optical flows; specifically, we discard optical-flow constraints at pixels where the sum of the forward and backward flows causes a displacement greater than 1 pixel.

6.2. Choosing the recovery frame rate. Before we detail the individual steps of the CS-MUVI video recovery procedure, it is important to specify the rate of the frames to be recovered. When sensing scenes with SMC architectures, there is no obvious notion of frame rate. One notion of the frame rate comes from the measurement rate, which in the case of the SPC is the operating rate of the DMD. However, this rate is extremely high and leads to videos whose dimensions are too high to allow feasible computations. Further, each frame would be associated with a *single* measurement, which leads to a severely ill-conditioned inverse problem. A potential definition comes from the work of Park and Wakin [26], who argue that the frame rate is not necessarily defined by the measurement rate. Specifically, the spatial bandwidth of the video often places an upper-bound on its temporal bandwidth as well. Intuitively, the idea here is that the larger the pixel size (or the smaller the spatial bandwidth), the greater the motion to register a change in the scene. Hence, given a scene motion in terms of pixels/second, a suitable notion of frame rate is one that ensures subpixel motion between consecutive frames. This notion is more meaningful since it intuitively weaves the observability of the motion into the definition of the frame rate. Under this definition, we wish to find the largest window size $\Delta W \leq W$ such that there is virtually no motion at full resolution $(n \times n)$. In practice, an estimate of ΔW can be obtained by analyzing the preview frames. Hence, given a total number of T compressive measurements, we ultimately recover $F = T/\Delta W$ full-resolution frames. Note that a smaller value of ΔW would decrease the amount of motion associated with each recovered frame; this would, however, increase the computational complexity (and memory requirements) substantially as the number of full-resolution frames to be recovered increases. Finally, the choice of ΔW is inherently scene-specific; scenes with fast moving highly textured objects require a smaller ΔW compared to those with slow moving smooth objects. The choice of ΔW could potentially be made time-varying as well and derived from the preview; this showcases the versatility of having the preview and is an important avenue for future research.

6.3. Recovery of full-resolution frames. We are now ready to detail the final stage of CS-MUVI. Assume that ΔW is chosen such that there is little to no motion associated with each preview frame. Next, associate a preview frame with a high-resolution frame $\mathbf{x}_k$, $k \in \{1, \ldots, T\}$, by grouping $W = N_L$ compressive measurements in the immediate vicinity of the frame (since $\Delta W \leq W$). Then, compute the optical flow between successive (upscaled) preview frames.

We can now recover the high-resolution video frames as follows. We enforce sparse spatio-temporal gradients using the 3D total variation (TV) norm. We furthermore consider the following two constraints: (i) consistency with the acquired CS measurements, i.e., $\mathbf{y}_t = \langle \phi_t, \mathbf{x}_{I(t)} \rangle$, where $I(t)$ maps the sample index t to the associated frame index k; and (ii) estimated optical-flow constraints between consecutive frames. Together, we arrive at the

following convex-optimization problem:

$$(\mathrm{TV}) \begin{cases} \text{minimize} & \mathrm{TV}_{3\mathrm{D}}(\mathbf{x}) \\ \text{subject to} & \left\|\langle \phi_t, \mathbf{x}_{I(t)} \rangle - y_t\right\|_2 \le \epsilon_1, \\ & \|\mathbf{x}_i(x,y) - \mathbf{x}_j(x+u_x, y+v_y)\|_2 \le \epsilon_2, \end{cases}$$

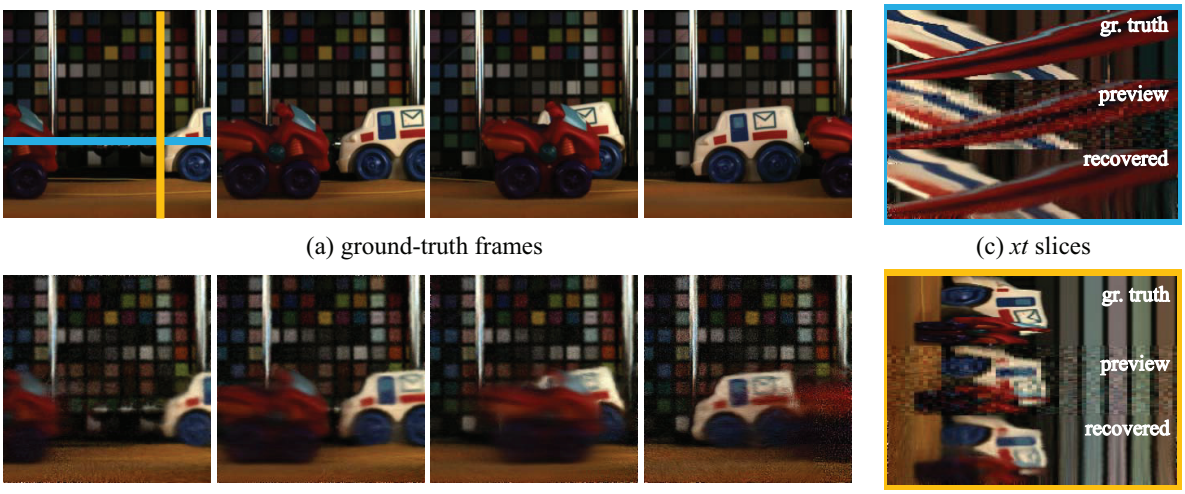
which can be solved using standard convex-optimization techniques. The specific technique that we employed was by variable splitting and using ALM/ADMM.

The parameters ϵ_1 and ϵ_2 are indicative of the measurement noise levels and the inaccuracies in the brightness constancy, respectively. ϵ_1 captures all sources of measurement noise including photon, dark, and read noise. Photon noise is signal dependent. However, in an SPC, each measurement is the sum of a random selection of half the micromirrors on the DMD. For most natural scenes, we can expect the measurements to be tightly clustered—to be more specific, around one-half of the total light level of the scene. Hence, the photon noise will have nearly the same variance across the measurements. Hence, for the SPC, all sources of measurement noise can be represented using one parameter ϵ_1 which is set via a calibration process. Setting ϵ_2 is based on the thresholds used in detecting violation of brightness constancy when estimating brightness constancy. For the results in this paper, ϵ_2 is set to $0.02 \times \sqrt{P}$, where P is the total number of pixel pairs for which we enforce brightness constancy.

7. Evaluation and comparisons. In this section, we validate the performance and capabilities of the CS-MUVI framework using simulations. Results on real data obtained from our SPC lab prototype are presented in section 8. All simulation results were generated from high-speed videos having a spatial resolution of $n \times n = 256 \times 256$ pixels. The preview videos have a spatial resolution of 64×64 pixels (i.e., $W = 4096$). We assume an SPC architecture as described in [8] with parameters chosen to mimic operation of our lab setup. Noise was added to the compressive measurements using an i.i.d. Gaussian noise model such that the resulting SNR was 60 dB. Optical-flow estimates were extracted using the method described in [19]. The computation time of CS-MUVI is dominated by both optical-flow estimation and solving (TV). Typical runtimes for the entire algorithm are 2–3 hours on an off-the-shelf quad-core CPU for a video of resolution of 256×256 pixels with 256 frames. However, computation of the low-resolution preview can be done almost instantaneously.

Video sequences from a high-speed camera. The results shown in Figures 9 and 10 correspond to scenes acquired by a high-speed video camera operating at 250 frames per second. Both videos show complex (and fast) movement of large objects as well as severe occlusions. For both sequences, we emulate an SPC operating at 8192 compressive measurements per second. For each video, we used 2048 frames of the high-speed camera to obtain a total of $T = 32 \times 2048$ compressive measurements. The final recovered video sequences consist of $F = 61$ frames ($\Delta W = 1024$). Both recovered videos demonstrate the effectiveness of CS-MUVI.

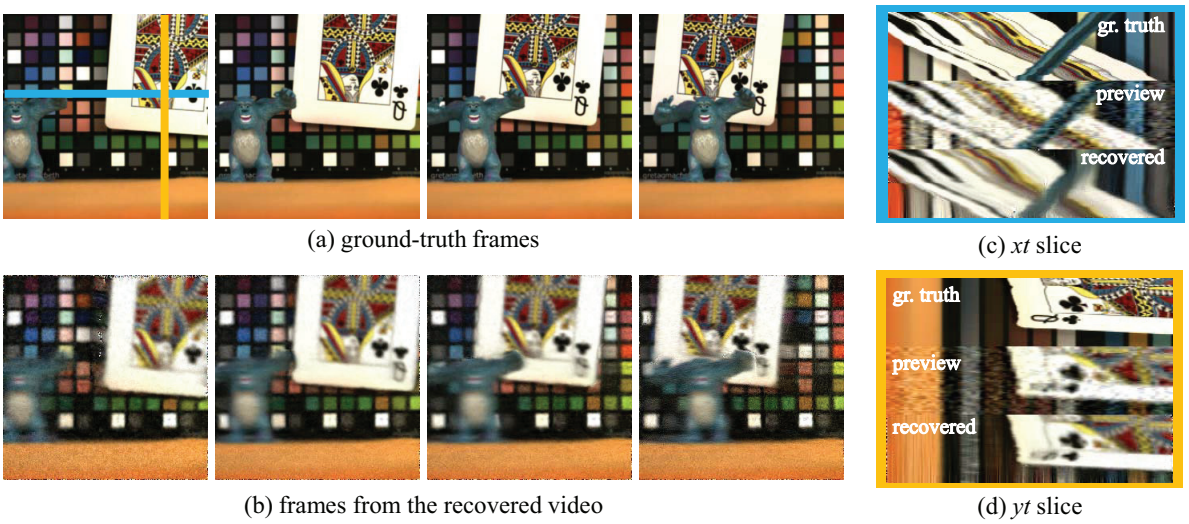
Comparison with the P2C2 algorithm. In the P2C2 camera [28], a two-step recovery algorithm—similar to CS-MUVI—is presented. This algorithm is nearly identical to CS-MUVI except that the measurement model does not use DSS measurement matrices; hence, an initial recovery using wavelet sparse models is used to obtain an initial estimate that plays



(a) ground-truth frames (c) *xt* slices

(b) frames from the recovered video (d) *yt* slice

Figure 9. Recovery on high-speed videos. *CS-MUVI recovery results of a video obtained from a high-speed camera operating at* 250 *fps (frames per second). Shown are frames of* (a) *the ground truth and* (b) *the recovered video (PSNR* = 25.0 *dB). The xt and yt slices shown in* (c) *and* (d) *correspond to the color-coded lines of the first frame in* (a). *Preview frames for this video are shown in Figure* 8. *(The xt and yt slices are rotated clockwise by* 90 *degrees.)*



(a) ground-truth frames (c) *xt* slice

(b) frames from the recovered video (d) *yt* slice

Figure 10. Recovery on high-speed videos. *CS-MUVI recovery results of a video obtained from a high-speed camera. Shown are frames of* (a) *the ground truth and* (b) *the recovered video (PSNR* = 20.4 *dB). The xt and yt slices shown in* (c) *and* (d) *correspond to the color-coded lines of the first frame in* (a). *Preview frames for this video are shown in Figure* 8. *(The xt and yt slices are rotated clockwise by* 90 *degrees.)*

the role of the preview frames. Figure 11 presents the results of both CS-MUVI and the recovery algorithm for the P2C2 camera [28], with the same number of measurements per compression level. It should be noted that the P2C2 camera algorithm was developed for

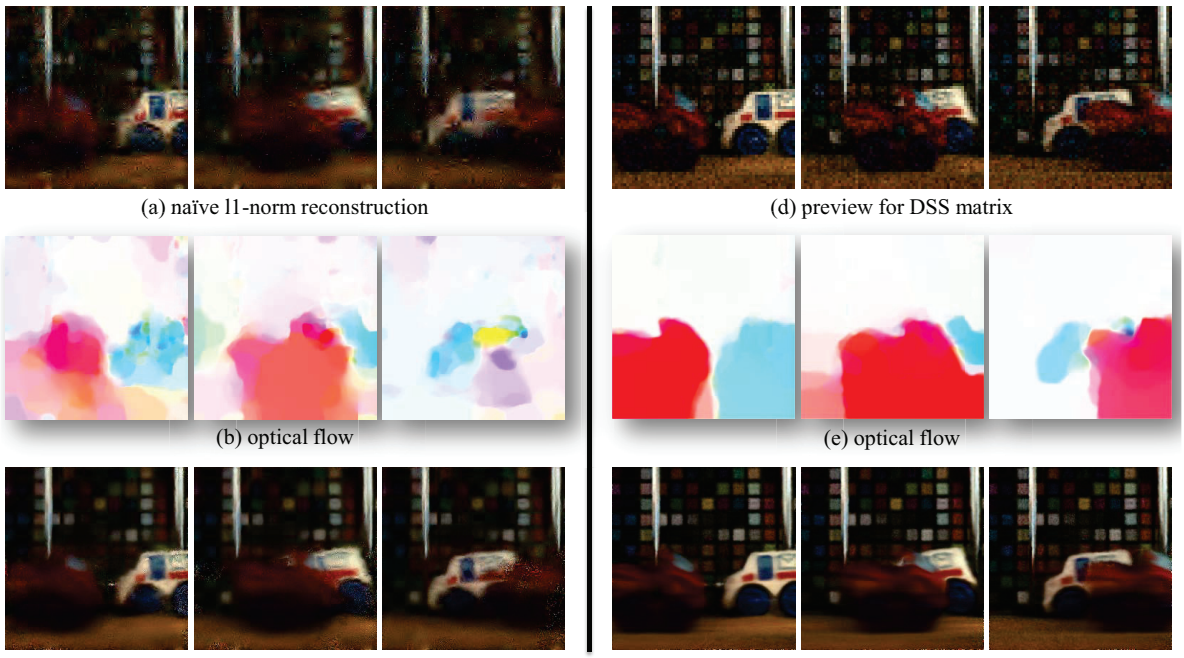
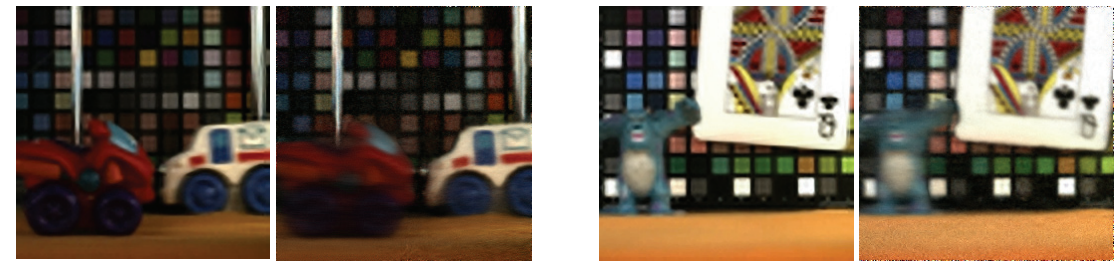


Figure 11. Comparisons to the two-step strategy used in the P2C2 camera [28]. *Shown are frames of* (a) *reconstruction obtained by minimizing the* ℓ_1*-norm of wavelet coefficients,* (b) *the resulting optical-flow estimates, and* (c) *the P2C2 recovered video. The frames in* (d) *correspond to preview frames when using DSS matrices,* (e) *are the optical-flow estimates, and* (f) *is the scene recovered by CS-MUVI.*

TMCs and *not* for SMC architectures. Nevertheless, we observe from Figures 11(a) and 11(d) that naïve ℓ_1-norm recovery delivers significantly worse initial estimates than the preview mode of CS-MUVI. The advantage of CS-MUVI for SMC architectures is also visible in the corresponding optical-flow estimates (see Figures 11(b) and 11(e)). The P2C2 recovery algorithm has substantial artifacts, whereas the result of CS-MUVI is visually pleasing. In all, this demonstrates the importance of the DSS matrix and the ability to robustly obtain a preview of the video.

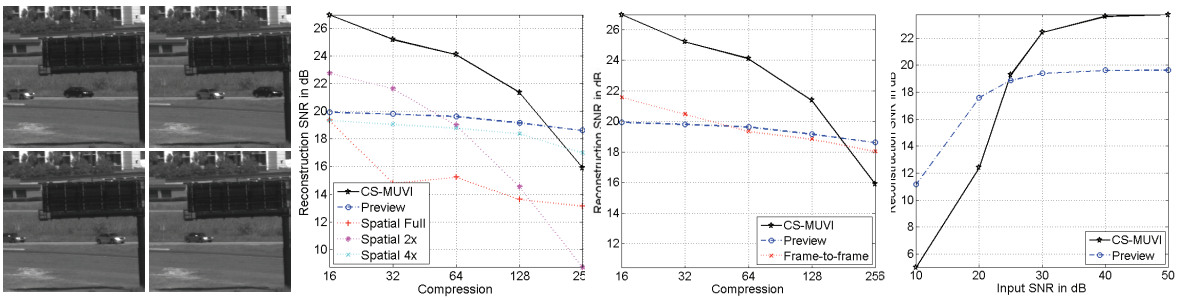
Comparisons against single-image super-resolution. There has been remarkable progress in single-image super-resolution. Figure 12 compares CS-MUVI to a sparse dictionary-based super-resolution algorithm [41]. From our observations, the results produced by the super-resolution are comparable to CS-MUVI when the upsampling is about 4×. However, in spite of this, the best known results in super-resolution seldom produce meaningful results beyond 4× super-resolution. Our proposed technique is in many ways similar to super-resolution except that we obtain multiple coded measurements of the scene, and this allows us to obtain higher super-resolution factors at potential loss in temporal resolution.

Performance analysis. Finally, we look at quantitative evaluation of CS-MUVI for varying compression ratios and input measurement noise level. Our metric for performance is



(a) Single-image super-resolution (b) CS-MUVI (c) Single-image super-resolution (d) CS-MUVI

Figure 12. Comparisons to the single-image super-resolution algorithm of [41]. *Shown are results on two high-speed videos.* (a), (c) *We use a low-resolution Hadamard matrix to sense a low-resolution image with* 64×64 *pixels and, subsequently, super-resolve them* $4\times$. (b), (d) *We use DSS matrices instead of low-resolution Hadamard to obtain the CS-MUVI results. Both algorithms have the same measurement rate. We observe that performance of CS-MUVI is similar to that of the super-resolution algorithm.*



(a) (b) (c) (d)

Figure 13. Quantitative performance. (a) *Four frames from a high-speed video.* (b) *Performance of CS-MUVI for different compression ratios compared against "Nyquist" cameras that trade-off spatial and temporal resolution to achieve the desired compression.* (c) *Performance of CS-MUVI compared against video recovered using frame-to-frame sparse wavelet prior. For the sparse wavelet prior, for each compression ratio, the window of measurements associated with each recovered frame was varied and the best performing result is shown.* (d) *Performance of CS-MUVI for varying levels of measurement noise. For high noise levels (low input SNR), the low quality preview leads to poor optical-flow estimates, which causes a severe degradation in performance.*

reconstruction SNR in dB defined as follows:

$$RSNR = -20 \log_{10} \left(\frac{\|\mathbf{x} - \widehat{\mathbf{x}}\|_2}{\|\mathbf{x}\|_2} \right),$$

where $\mathbf{x}$ and $\widehat{\mathbf{x}}$ are the ground truth and estimated video, respectively. The test-data for this is a 250 fps video of vehicles on a highway. A few frames from this video are shown in Figure 13(a). We establish a baseline for these results using two different algorithms. First, we consider "Nyquist cameras" that blindly trade-off spatial and temporal resolution to achieve the desired compression. For example, at a compression factor of $16\times$, a Nyquist camera could deliver full resolution at 1/16th the temporal resolution or deliver 1/2th the spatial resolution at 1/8th the temporal resolution, and so on. This spatio-temporal trade-off is feasible in most traditional imagers by binning pixels at readout. Second, we consider videos recovered using naïve frame-to-frame wavelet priors. For such reconstructions, we optimized over different

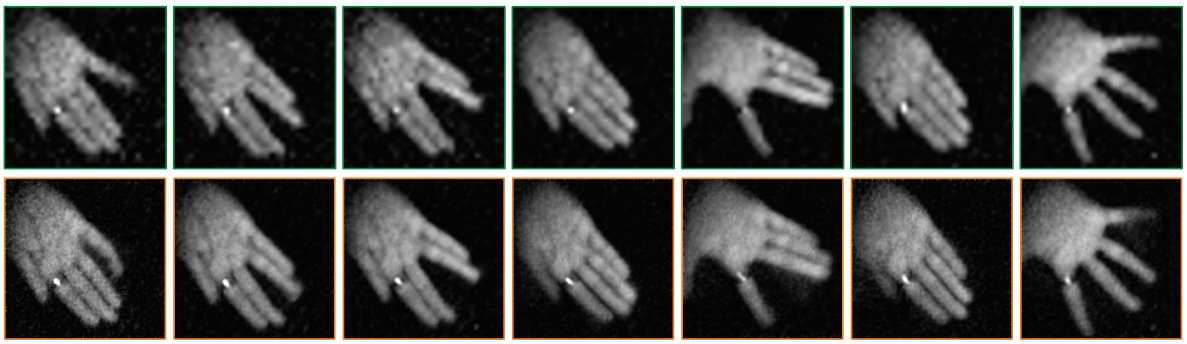
window lengths of measurements associated with each recovered frame and chose the setting that provided the best results. Figures 13(b),(c) show reconstruction SNR for CS-MUVI and the two baseline algorithms for varying levels of compression. At high compression ratios, the performance of CS-MUVI suffers from poor optical-flow estimates. Finally, in Figure 13(d), we present performance for varying levels of measurement or input noise. Again, as before, for high noise levels, optical-flow estimates suffer, leading to poorer reconstructions. In all, CS-MUVI delivers high-quality reconstructions for a wide range of compression and noise levels.

8. Hardware implementation. We now present video recovery results on real data from our SPC lab prototype.

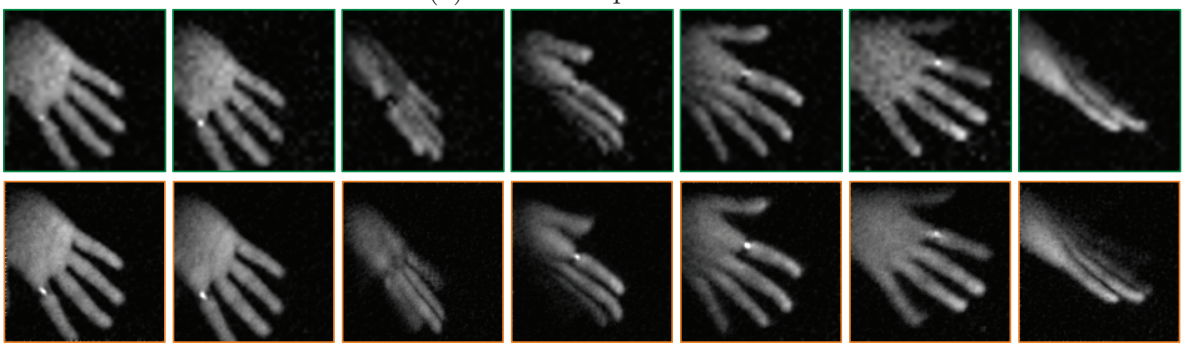
Hardware prototype. The SPC setup we used to image real scenes is comprised of a DMD operating at 10,000 mirror-flips per second. The real measured data were acquired using a SWIR photodetector for the scenes involving the pendulum and a visible photodetector for the rest (the hand and windmill scenes). While the DMD we used is capable of imaging the scene at an XGA resolution (i.e., 1024×768 pixels), we operate it at a lower spatial resolution, mainly for two reasons. First, recall that the measurement bandwidth of an SPC is determined by the speed of operation of the DMD. In our case, this was 10,000 measurements per second. Even if we were to obtain a compression of $50\times$, our device would be similar to a conventional sampler whose measurement bandwidth is 5×10^5 measurements/second, which would result in a video of approximately 128×128 pixels at 30 frames/second. Hence, we operate it at a spatial resolution of 128×128 pixels by grouping pixels together on the DMD as one 6×6 super-pixel. Second, the patterns displayed on the DMD were required to be preloaded onto the memory board attached to the DMD via a USB port. With limited memory, typically 96 GB, any reasonable temporal resolution with XGA resolution would be infeasible on our current SPC prototype. We emphasize that both of these are limitations due to the used prototype and not of the underlying algorithms. Recent, commercial DMDs can operate at least 1-to-2 orders of magnitude faster [23], and the increase in measurement bandwidth would enable sensing at higher spatial and temporal resolutions.

Gallery of real data results. Figure 14 shows a few example reconstructions from our SPC lab setup. Each video is approximately 1.6 seconds long and corresponds to $M = 16{,}384$ measurements from the SPC. With $D = 4$, all previews (the top row in each subimage in Figure 14) were of size 32×32 pixels. Videos were recovered with $F = 125$ frames. See the supplementary material for videos for each of the results (98312_01.pptx [local/web 68.2MB]).

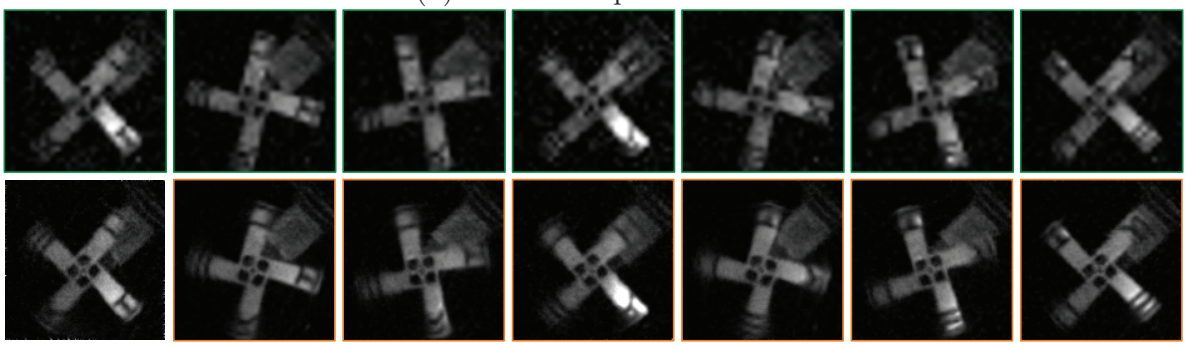
Role of different signal priors. Figures 2, 15, and 16 show the performance of three different signal priors on the same set of measurements. In Figure 2, we compare wavelet sparsity of the individual frames, 3D TV, and CS-MUVI, which uses optical-flow constraints in addition to the 3D TV model. CS-MUVI delivers superior performance in recovery of the spatial statistics (the textures on the individual frames) as well as temporal statistics (the textures on temporal slices). In Figure 15, we look at specific frames across a wide gamut of reconstructions where the target motion is very high. Again, we observe that reconstruction from CS-MUVI is not just free from artifacts; it also resolves spatial features better (ring on the hand, palm lines, etc.). Finally, for completeness, in Figure 16, we vary the number of measurements associated with each frame for both 3D TV and CS-MUVI. Predictably, while the performance of 3D



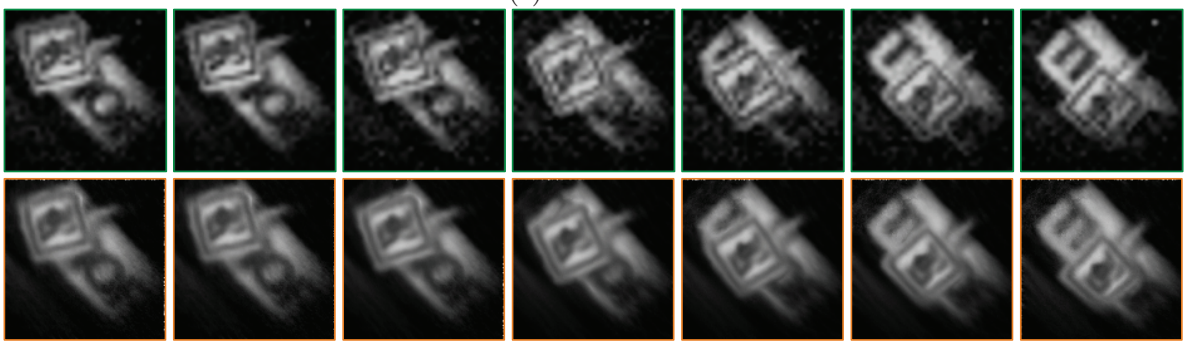
(a) Hand: Simple motion.



(b) Hand: Complex motion.

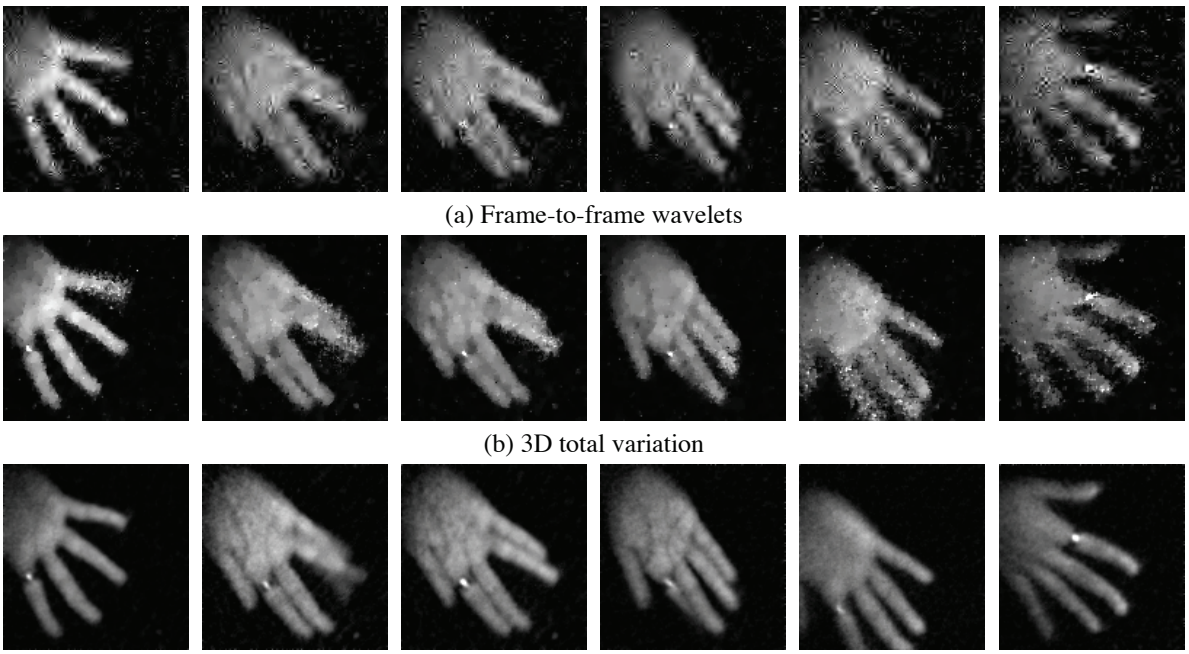


(c) Windmill.



(d) Pendulum.

Figure 14. Reconstructions from SPC hardware. (a)–(d) *show four different scenes with different kinds of motion. For each scene, the top row (marked in green) shows frames from the preview, and the bottom row (red) shows the corresponding frames from the final recovered video.*



(a) Frame-to-frame wavelets

(b) 3D total variation

(c) CS-MUVI

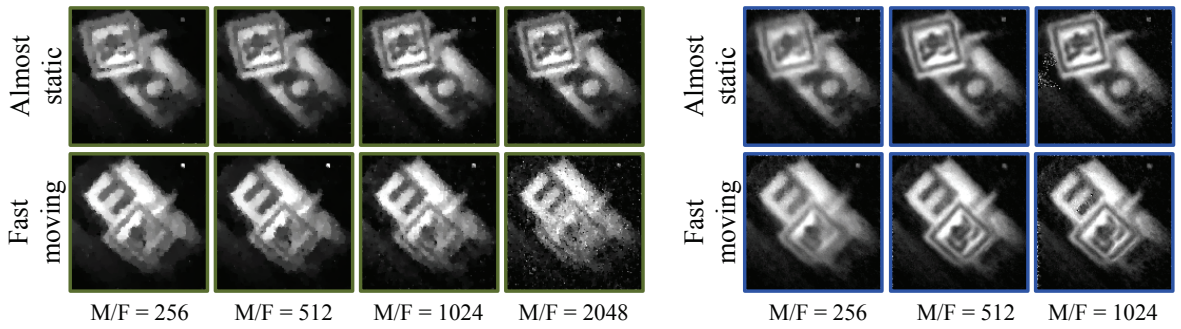
Figure 15. Performance comparison of different signal models. *We look at performance of various signal models for a dynamic, fast moving target. Shown are select frames where the speed of the target was high. As before, CS-MUVI handles fast moving targets gracefully without any of the artifacts present in competing signal models. We refer the reader to the supplementary material for a complete video (*98312_01.pptx [local/web 68.2MB]*).*

TV is poor for fast moving objects, CS-MUVI delivers high-quality reconstructions across a wide range of target motion.

Achieved spatial resolution. In Figures 17 and 18, note that an SMC seeks to super-resolve a low-resolution sensor using optical coding and SLMs. Hence, it is of utmost importance to verify whether the device actually delivers on the promised improvement in spatial resolution.

In Figure 17, we present reconstruction results on a resolution chart. The resolution chart was translated so as to enter and exit the field-of-view of the SPC within 8 seconds providing a total of 86,000 measurements. A video with 159 frames was recovered from these measurements for an overall compression ratio of 32×. Figure 17 indicates that the CS-MUVI recovers spatial detail to a per-pixel precision, validating the claims of achieved compression. For this result, we regularized the optical flow to be translational. Specifically, after estimating the flow between the preview frames, we used the median of the flow-vectors as a global translational flow.

In Figure 18, we characterize the spatial resolution achieved by CS-MUVI by comparing it to the image of a static scene obtained using pure Hadamard multiplexing. As expected, we observe that the preview image is the same resolution as the static image downsampled 4×. Frames recovered from CS-MUVI exhibit sharper texture than a 2× downsampling of the static frame but slightly worse texture than the full-resolution static image. Note that this scene contained complex nonrigid and fast motion.



(a) 3D TV without optical flow constraints (b) 3D TV with optical flow constraints

Figure 16. Comparison of recovered videos with and without optical-flow constraints. *Data were collected with an SPC operating at* 10,000 *Hz with an SWIR photodetector. A total of* $M = 16{,}384$ *compressive measurements were obtained at a DMD resolution of* 128×128. *In each case, we show multiple reconstructions with a different number of compressive measurements associated with each frame. That is, in each instance, the number of recovered frames* F *is chosen to satisfy the target* M/F *value.* (a) *Reconstructions without optical-flow constraints. The top row shows the pendulum at one end of its swing where it is nearly stationary. The bottom row shows the pendulum when it is moving the fastest. As expected, increasing the number of measurements per frame,* M/F, *increases the motion blur significantly.* (b) *In contrast, the use of optical flow preserves the quality of the results. The visual quality peaks at* $M/F = 512$ *(see the supplementary material for videos (*98312_01.pptx [local/web 68.2MB]*)).*

Variations in speed, illumination, and size. Finally, we look at performance on real data for varying levels of scene illumination, object speed, and size. For illumination (Figure 19), we use the SPC measurement level as a guide to the amount of scene illumination. For object speed (Figure 20), we instead slow down the DMD since it indirectly provides finer control on the apparent speed of the object. For size (Figure 21), we vary the size of the moving target. In all cases, we show the recovered frame corresponding to the object moving at the fastest speed. The performance of CS-MUVI degrades gracefully across all variations. The interested reader is referred to the supplementary material for videos of these results (98312_01.pptx [local/web 68.2MB]).

9. Discussion.

Summary. The promise of an SMC is to deliver high spatial resolution images and videos from a low-resolution sensor. The most extreme form of such SMCs is the SPC, which poses a single photodetector or a sensor with no resolution by itself. In this paper, we demonstrate—for the very first time on real data—successful video recovery at 128× super-resolution for fast-moving scenes. This result has important implications for regimes where high-resolution sensors are prohibitively expensive. A example of this is imaging in SWIR; to this end, we show results using an SPC with a photodetector tuned to this spectral band.

At the heart of our proposed framework is the design of a novel class of sensing matrices and an optical flow–based video reconstruction algorithm. In particular, we have proposed dual-scale sensing (DSS) matrices that (i) exhibit no noise enhancement when performing least-squares estimation at low spatial resolution and (ii) preserve information about high spatial frequencies. We have developed a DSS matrix having a fast transform, which enables

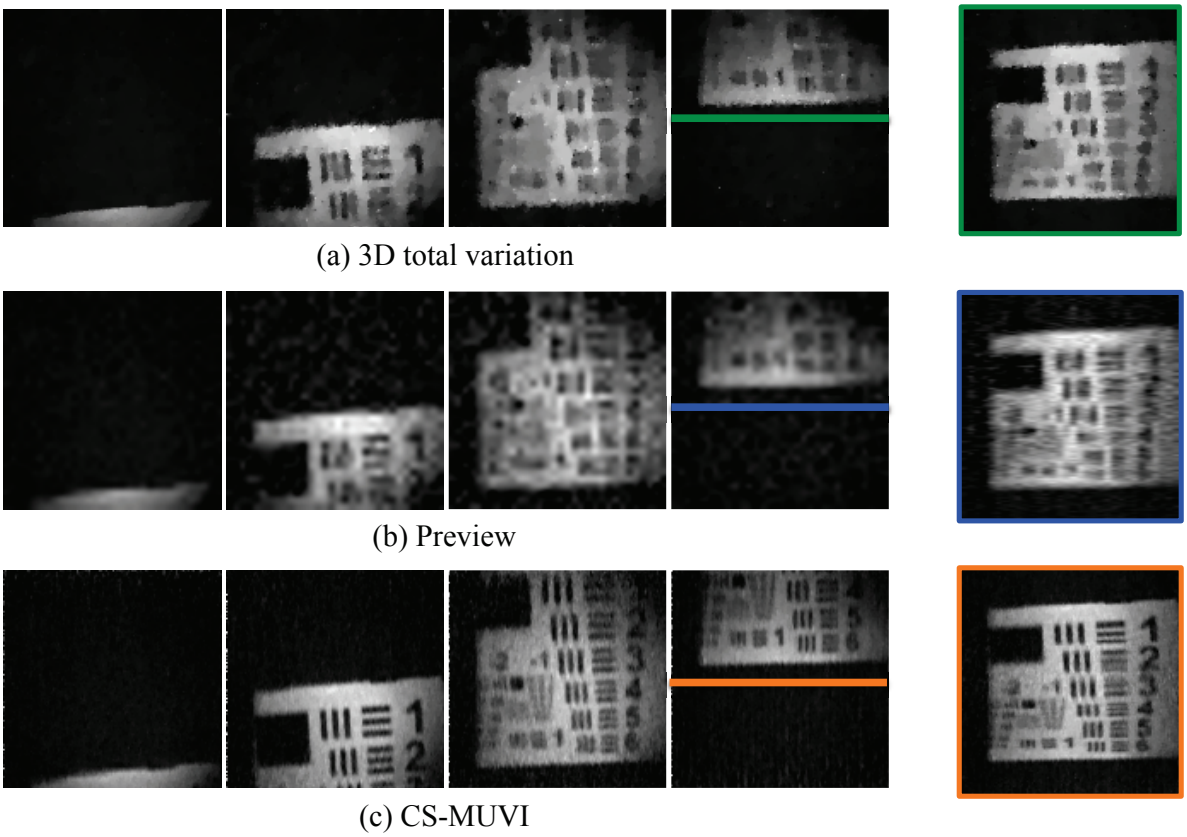


Figure 17. Resolution chart. *Reconstruction results on a translating resolution chart at a compression ratio of* $32\times$. *In each row, we show frames from the recovered video as well as its xt slice in the color-coded box in the last column.*

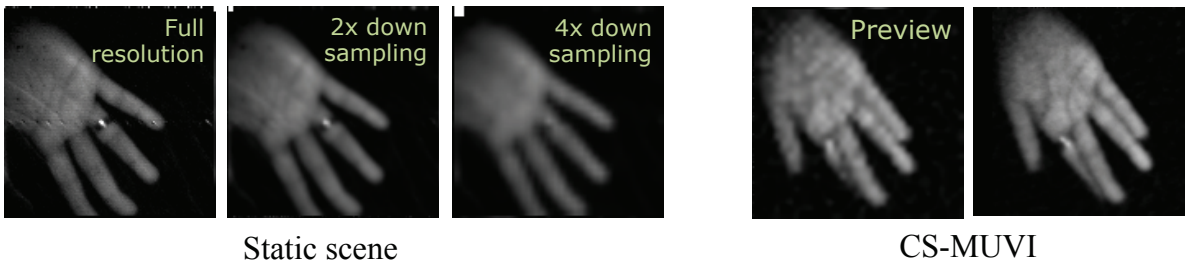


Figure 18. Achieved resolution. *We compare the achieved spatial resolution of the recovered video for a static target. For visual comparison, we artificially downsample the static image. It is clear that CS-MUVI recovers spatial resolution higher than a* $2\times$ *downsampling but slightly worse than the full-resolution image.*

us to compute instantaneous *preview* images of the scene at low cost. The preview computation supports a large number of novel applications for SMC-based devices, such as providing a digital viewfinder, enabling human-camera interaction, or triggering adaptive sensing strategies.

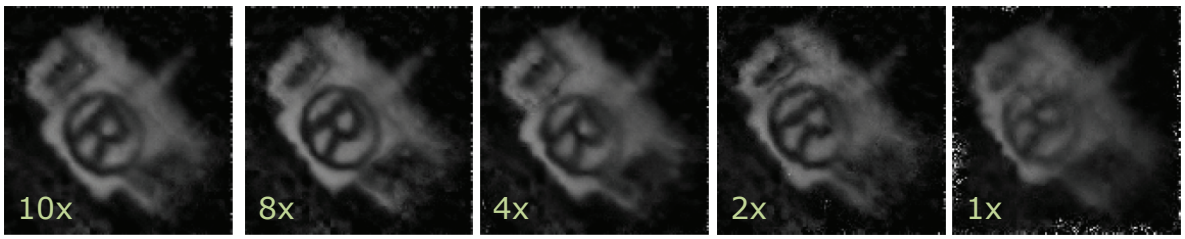


Figure 19. Performance for varying scene illumination levels. *We controlled the total light level in the scene by controlling the light throughput of the illumination sources. Shown are results at different scene light levels—each case calibrated by the multiple of the minimum light level. In each case, we show one frame of the recovered video, the instant corresponding to the pendulum swinging at maximum speed. The performance degradation of the algorithm is graceful with only little artifacts.*

Figure 20. Performance for varying speed. *We slowed down the operating speed of the SPC to indirectly increase object speed. The operating speed of the SPC is overlaid on top of the recovered video. Shown is a single frame from each recovered video, the instant corresponding to the pendulum swinging at maximum speed.*

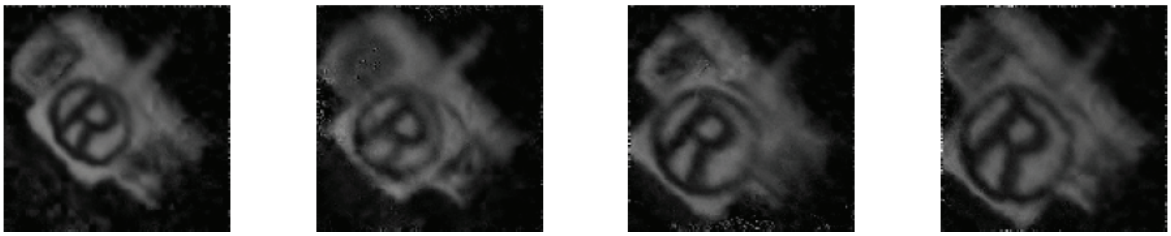


Figure 21. Performance for varying size of dynamic object. *For a wide range of object size, from a quarter to half of the entire field-of-view of the camera, we obtain stable reconstructions.*

Limitations. Since CS-MUVI relies on optical-flow estimates obtained from low-resolution images, it can fail to recover small objects with rapid motion. More specifically, moving objects that are of subpixel size in the preview mode are lost. Figure 9 shows an example of this limitation: the cars are moved using fine strings, which are visible in Figure 9(a) but not in Figure 9(b). Increasing the spatial resolution of the preview images eliminates this problem at the cost of more motion blur. To avoid these limitations altogether, one must increase the sampling rate of the SMC. In addition, reducing the complexity of solving (TV) is of paramount importance for practical implementations of CS-MUVI.

Faster implementations. Current implementation of CS-MUVI takes in the order of hours for high-resolution videos with a large number of frames. This large runtime can be attributed

to the DSS matrix lacking a fast transform as well as the inherent complexity associated with high-resolution signals. Faster implementation of the recovery algorithm is an interesting research direction.

Multiscale preview. A drawback of our approach is the need to specify the resolution at which preview frames are recovered; this requires prior knowledge of object speed. An important direction for future work is to relax this requirement via the construction of multiscale sensing matrices that go beyond the DSS matrices proposed here. The recently proposed sum-to-one (STOne) transform [11] provides such a multiscale sensing matrix. Specifically, the STOne transform is a carefully designed Hadamard transform that remains a Hadamard transform of a lower resolution when downsampled. Using the STOne transform in place of the DSS matrix could potentially provide previews of various spatial resolutions.

Multiframe optical flow. The majority of the artifacts in the reconstructions stem from inaccurate optical-flow estimates—a result of residual noise in the preview images. It is worth noting, however, that we are using an off-the-shelf optical-flow estimation algorithm; such an approach ignores the continuity of motion across *multiple* frames. We envision significant performance improvements if we use multiframe optical-flow estimation [30]. Such an approach could potentially alleviate some of the challenges faced in pairwise optical flow, including the inability to recover precise flow estimates for both slow-moving and fast-moving targets.

Towards high-resolution imagers. The spatial resolution of an SMC is limited by the resolution of the SLM. Commercially available DMDs, LCDs, and LCOSs have a spatial resolution of 1–2 megapixels. An important direction for future research is the design of imaging architectures, signal models, and recovery algorithms to obtain videos at this spatial resolution (and say, 30 fps temporal resolution). The key stumbling block for an SPC-based approach for solving this is the measurement bandwidth which, for the SPC, is limited by the operating rate of DMD. An approach to increasing the measurement rate is to use a multipixel architecture [6, 21, 38]. One way to interpret such imagers is to think of each pixel on the sensor as an SPC. Hence, with the successful 128× demonstrated in this paper, megapixel videos could potentially be achieved with the use of an 8×8 photodetector array. However, the very high dimensionality of the recovered videos raises important computational challenges with regard to the use of optical flow–based recovery algorithms.

REFERENCES

[1] M. Ben-Or and P. Tiwari, *A deterministic algorithm for sparse multivariate polynomial interpolation*, in Proceedings of the 20th Annual ACM Symposium on Theory of Computing, ACM, New York, 1988, pp. 301–309.

[2] E. J. Candès, *The restricted isometry property and its implications for compressed sensing*, C. R. Math. Acad. Sci. Paris, 346 (2008), pp. 589–592.

[3] E. J. Candès, J. Romberg, and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inf. Theory, 52 (2006), pp. 489–509.

[4] V. Cevher, A. C. Sankaranarayanan, M. F. Duarte, D. Reddy, R. G. Baraniuk, and R. Chellappa, *Compressive sensing for background subtraction*, in Proceedings of the European Conference on Computer Vision (ECCV), Marseille, France, 2008, pp. 155–168.

[5] A. Chambolle, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vision, 20 (2004), pp. 89–97.

[6] H. Chen, M. S. Asif, A. C. Sankaranarayanan, and A. Veeraraghavan, *FPA-CS: Focal plane array-based compressive imaging in short-wave infrared*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015.
[7] D. L. Donoho, *Compressed sensing*, IEEE Trans. Inf. Theory, 52 (2006), pp. 1289–1306.
[8] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, *Single-pixel imaging via compressive sampling*, IEEE Signal Process. Mag., 25 (2008), pp. 83–91.
[9] J. E. Fowler, S. Mun, E. W. Tramel, M. R. Gupta, Y. Chen, T. Wiegand, and H. Schwarz, *Block-based compressed sensing of images and video*, Found. Trends Signal Process., 4 (2010), pp. 297–416.
[10] M. E. Gehm and D. J. Brady, *Compressive sensing in the EO/IR*, Appl. Opt., 54 (2015), pp. C14–C22.
[11] T. Goldstein, L. Xu, K. F. Kelly, and R. G. Baraniuk, *The STOne Transform: Multi-resolution Image Enhancement and Real-Time Compressive Video*, preprint, arXiv:1311.3405, 2013.
[12] J. Gu, Y. Hitomi, T. Mitsunaga, and S. Nayar, *Coded rolling shutter photography: Flexible space-time sampling*, in Proceedings of the IEEE International Conference on Computational Photography (ICCP), Cambridge, MA, 2010, pp. 1–8.
[13] Z. T. Harmany, R. F. Marcia, and R. M. Willett, *Compressive Coded Aperture Keyed Exposure Imaging with Optical Flow Reconstruction*, preprint, arXiv:1306.6281, 2013.
[14] M. Harwit and N. J. Sloane, *Hadamard Transform Optics*, Academic Press, New York, 1979.
[15] Y. Hitomi, J. Gu, M. Gupta, T. Mitsunaga, and S. K. Nayar, *Video from a single coded exposure photograph using a learned over-complete dictionary*, in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, 2011, pp. 287–294.
[16] J. Holloway, A. C. Sankaranarayanan, A. Veeraraghavan, and S. Tambe, *Flutter shutter video camera for compressive sensing of videos*, in Proceedings of the IEEE International Conference on Computational Photography (ICCP), Seattle, WA, 2012, pp. 1–9.
[17] B. K. P. Horn and B. G. Schunck, *Determining optical flow*, Artif. Intel., 17 (1981), pp. 185–203.
[18] D. Le Gall, *MPEG: A video compression standard for multimedia applications*, Comm. ACM, 34 (1991), pp. 46–58.
[19] C. Liu, *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 2009.
[20] P. Llull, X. Liao, X. Yuan, J. Yang, D. Kittle, L. Carin, G. Sapiro, and D. J. Brady, *Coded aperture compressive temporal imaging*, Opt. Express, 21 (2013), pp. 10526–10545.
[21] A. Mahalanobis, R. Shilling, R. Murphy, and R. Muise, *Recent results of medium wave infrared compressive sensing*, Appl. Opt., 53 (2014), pp. 8060–8070.
[22] S. Mun and J. E. Fowler, *Residual reconstruction for block-based compressed sensing of video*, in Proceedings of the IEEE Conference on Data Compression, Snowbird, UT, 2011, pp. 183–192.
[23] S. G. Narasimhan, S. J. Koppal, and S. Yamazaki, *Temporal dithering of illumination for fast active vision*, in Proceedings of the 10th European Conference on Computer Vision (ECCV), Marseille, France, Lecture Notes in Comput. Sci. 5305, Springer, Berlin, 2008, pp. 830–844.
[24] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An iterative regularization method for total variation-based image restoration*, Multiscale Model. Simul., 4 (2005), pp. 460–489.
[25] J. Y. Park and M. B. Wakin, *A multiscale framework for compressive sensing of video*, in Proceedings of the IEEE Picture Coding Symposium, Chicago, IL, 2009, pp. 197–200.
[26] J. Y. Park and M. B. Wakin, *Multiscale algorithm for reconstructing videos from streaming compressive measurements*, J. Electron. Imaging, 22 (2013), 021001.
[27] R. Raskar, A. Agrawal, and J. Tumblin, *Coded exposure photography: Motion deblurring using fluttered shutter*, ACM Trans. Graphics, 25 (2006), pp. 795–804.
[28] D. Reddy, A. Veeraraghavan, and R. Chellappa, *P2C2: Programmable pixel compressive camera for high speed imaging*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, 2011, pp. 329–336.
[29] I. E. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia*, John Wiley and Sons, New York, 2004.
[30] M. Rubinstein, C. Liu, and W. T. Freeman, *Towards longer long-range motion trajectories*, in Proceedings of the British Machine Vision Conference, BMVA Press, Durham, UK, 2012, pp. 53.1–53.11.

[31] A. C. Sankaranarayanan, C. Studer, and R. G. Baraniuk, *CS-MUVI: Video compressive sensing for spatial-multiplexing cameras*, in Proceedings of the IEEE International Conference on Computational Photography (ICCP), Seattle, WA, 2012, pp. 1–10.

[32] A. C. Sankaranarayanan, P. Turaga, R. Baraniuk, and R. Chellappa, *Compressive acquisition of dynamic scenes*, in Proceedings of the 11th European Conference on Computer Vision (ECCV), Crete, Greece, Lecture Notes in Comput. Sci. 6311, Springer, Berlin, 2010, pp. 129–142.

[33] A. C. Sankaranarayanan, P. K. Turaga, R. Chellappa, and R. G. Baraniuk, *Compressive acquisition of linear dynamical systems*, SIAM J. Imaging Sci., 6 (2013), pp. 2109–2133.

[34] N. Vaswani, *Kalman filtered compressed sensing*, in Proceedings of the 15th IEEE Conference on Image Processing (ICIP), San Diego, CA, 2008, pp. 893–896.

[35] N. Vaswani and W. Lu, *Modified-CS: Modifying compressive sensing for problems with partially known support*, IEEE Trans. Signal Process., 58 (2010), pp. 4595–4607.

[36] A. Veeraraghavan, D. Reddy, and R. Raskar, *Coded strobing photography: Compressive sensing of high speed periodic events*, IEEE Trans. Pattern Anal. Mach. Intell., 33 (2011), pp. 671–686.

[37] M. B. Wakin, J. N. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly, and R. G. Baraniuk, *Compressive imaging for video representation and coding*, in Proceedings of the 25th Picture Coding Symposium, Beijing, China, 2006, pp. 711–716.

[38] J. Wang, M. Gupta, and A. C. Sankaranarayanan, *LiSens—A scalable architecture for video compressive sensing*, in Proceedings of the IEEE Conference on Computational Photography (ICCP), Houston, TX, 2015.

[39] A. E. Waters, A. C. Sankaranarayanan, and R. G. Baraniuk, *SpaRCS: Recovering low-rank and sparse matrices from compressive measurements*, in Proceedings of the 24th Advances in Neural Information Processing Systems (NIPS), Granada, Spain, 2011, pp. 1089–1097.

[40] L. Xu, A. Sankaranarayanan, C. Studer, Y. Li, R. G. Baraniuk, and K. F. Kelly, *Multiscale compressive video acquisition*, in Proceedings of Computational Optical Sensing and Imaging, Alexandria, VA, 2013, CW2C.4.

[41] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, *Coupled dictionary training for image super-resolution*, IEEE Trans. Image Process., 21 (2012), pp. 3467–3478.

[42] J. Yang, X. Yuan, X. Liao, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, *Video compressive sensing using Gaussian mixture models*, IEEE Trans. Image Process., 23 (2014), pp. 4863–4878.